Rapid Construction of Safe Search-Trees for Spacecraft Attitude Planning

Claus Danielson[†], Joseph Kloeppel[‡]

***Abstract*— This paper adapts the rapidly-exploring variant of invariant-set motion planner (ISMP) for spacecraft attitude motion planning and control. The ISMP is a motion-planning algorithm that uses positive-invariant sets of the closed-loop dynamics to find a constraint admissible path to a desired target through an obstacle filled environment. We present four mathematical results that enable the sub-routines used to rapidly construct a search-tree for the ISMP. These mathematical results describe how to uniformly sample safe quaternions, how to find the nearest orientation in the search-tree, how to move the sampled orientation to form an edge, and how to scale the invariant set to guarantee constraint admissibility. We present simulation results that demonstrate the ISMP for spacecraft attitude motion planning.**

I. Introduction

The invariant-set motion-planner (ISMP) is motion planning algorithm that generates a sequence of references that safely guides a closed-loop system from an initial state to a target equilibrium through an obstacle-filled environment [1]–[10]. The defining feature of the ISMP is that knowledge of the closed-loop system dynamics is incorporated into the search graph using constraint admissible positive invariant (CAPI) sets (also called viable sets [11]). These CAPI sets describe regions of the state-space where the closed-loop system can safely track the corresponding references. The ISMP uses a graph search to find a corridor of safe sets that safely guides the system through the obstacle filled environment to the target equilibrium.

The ISMP has several beneficial properties. It allows for aggressive, but safe maneuvers since, by definition, the system state will never leave the CAPI sets. It is inherently robust since it incorporates feedback into the design and the CAPI sets provide a natural buffer that can absorb tracking errors due to model uncertainty and disturbances [4]. It typically has low online computational costs since the CAPI sets can be pre-computed as they only depend on the time-invariant closed-loop dynamics, rather than the time varying environment. Furthermore, it does not require dense sampling since the CAPI sets can cover large volumes of the state/output-space. In addition, it reduces the curse-of-dimensionality by sampling from the output-space instead of the state-space. In the case of spacecraft, this means we can sample orientation-space (quaternions) rather than the full state-space, which also includes the spacecraft angular velocity.

This material is based upon work supported by the National Science Foundation under NSF Grant Number CMMI-2105631 and the Air Force Office of Scientific Research under award number FA9550-22-1-0093. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation nor the United States Air Force. [†]cdanielson@unm.edu, Assistant Professor, Department of Mechanical Engineering, University of New Mexico [‡]Air Force Research Laboratory AFRL/VSSV OrgMailbox@us.af.mil

We previously applied the ISMP for spacecraft attitude control in [6]. That paper used a pre-constructed search-graph where unsafe nodes were removed online if they collided with a keep-out cone. This required a dense uniform graph to increase the likelihood of finding a path through an arbitrary arrangement of keep-out cones. The main contribution of that paper was a computationally efficient method for testing the safety of nodes. In contrast, this paper focuses on the online construction of a search-tree that is tailored to the arrangement of keep-out cones. This requires fewer nodes and can potentially produce more desirable paths.

The rapid-exploring variant of the ISMP [7] combines the advantages of the ISMP [1]–[10] with the rapidly-exploring random tree (RRT) algorithm [12], [13]. The rapid-ISMP constructs a search-tree online using the sampling procedure from RRT. The main-loop of the algorithm samples a point from the output space and then moves this sample to form a connection with the nearest point in the search-tree. The rapid-ISMP has an additional step where a CAPI set is constructed around the newly sampled reference. This paper adapts the rapid-ISMP for spacecraft motion planning and control. We present four mathematical results that describe the sub-routines of the online search-tree construction algorithm. We describe how to uniformly sample [14] quaternion-representations of spacecraft orientations that satisfy keep-in cone constraints. Then, we describe how to find the nearest orientation in terms of the Minkowski function [15] of the CAPI set. Next, we describe how to move the sampled orientation using spherical linear interpolation (SLERP) so that it lies in the interior of the nearest CAPI set. Finally, we describe how to construct a CAPI set around the new reference orientation. We adapt the necessary and sufficient conditions from [6] to find the largest positive invariant (PI) set that is constraint admissible. The contributions of this paper are summarized:

i. A method for uniformly sampling quaternions from keep-in cones
ii. A closed-form expression for the Minkowski distance between orientations
iii. A closed-form for the SLERP connecting orientations
iv. A method for scaling PI sets for constraint admissibility

Notation and Definitions: A set $\mathcal{O}$ is positive invariant if $x(t_0) \in \mathcal{O} \Rightarrow x(t) \in \mathcal{O}\ \forall t > t_0$. Level-sets $\{x : V(x) \leq l\}$ of Lyapunov functions are positive invariant. $\mathbb{SO}(3) \subset \mathbb{R}^{3\times 3}$ is the group of rotation matrices i.e. $R^\top R = RR^\top = I$ and $\det(R) = +1$. With abuse of terminology, we will use $\mathbb{SO}(3)$ to refer to an abstract group isomorphic to $\mathbb{SO}(3)$. $\mathbb{SE}(3) = \mathbb{SO}(3) \times \mathbb{R}^3$. $\mathbb{S}^n = \{x \in \mathbb{R}^{n+1} : x^\top x = 1\}$.

The quaternions $q = (q_0, q_x, q_y, q_z) \in \mathbb{H}$ are a group of hyper-complex numbers with the Hamilton product $q \otimes p = (q_0 p_0 - \vec{q}^\top \vec{p}, q_0 \vec{p} + p_0 \vec{q} + \vec{q} \times \vec{p})$ where $\vec{q} = (q_x, q_y, q_z)$ and q_0 are the vector and scalar parts of q, respectively. $\mathbb{1} = (1, 0, 0, 0) \in \mathbb{H}$ is the identity quaternion. $\bar{q} = (q_0, -\vec{q})$ is the conjugate of $q = (q_0, \vec{q})$. $\bar{\mathbb{H}} = \mathbb{H} \cap \mathbb{S}^3$ denotes the unit quaternions. The unit quaternions $\bar{\mathbb{H}}$ are a double-cover of $\mathbb{SO}(3)$ since $\pm q \in \bar{\mathbb{H}}$ represent the same element of $\mathbb{SO}(3)$. With abuse of notation, $q \otimes v$ is the quaternion product of $q \in \mathbb{H}$ and $(0, v) \in \mathbb{H}$ where $v \in \mathbb{R}^3$. A directed graph $\mathfrak{G} = (\mathfrak{I}, \mathfrak{E})$ is a set of nodes $\mathfrak{I}$ together with a set of ordered pairs $\mathfrak{E} \subseteq \mathfrak{I} \times \mathfrak{I}$ called edges. Nodes $i, j \in \mathfrak{I}$ are called adjacent if $(i, j) \in \mathfrak{E}$ is an edge. A path is a sequence of adjacent vertices. A graph search is an algorithm for finding a path through a graph. A graph $\mathfrak{T}$ is a tree if every pair of nodes $(i, j) \in \mathfrak{I}$ is connected by exactly one path.

II. Attitude Planning Problem and Algorithm

A. Spacecraft Attitude Dynamics

The spacecraft attitude dynamics are modeled by the quaternion kinematics and Euler's equation, respectively [6]

$$\dot{q}(t) = \tfrac{1}{2} q(t) \otimes \omega(t) \tag{1a}$$
$$J\dot{\omega}(t) = -\omega(t) \times J\omega(t) + \tau(t) \tag{1b}$$

where the state $x = (q, \omega) \in \mathbb{SE}(3)$ is comprised of the spacecraft orientation $q \in \bar{\mathbb{H}}$ and angular velocity $\omega \in \mathbb{R}^3$ and the control input is the torque $\tau \in \mathbb{R}^3$. The parameter $J \in \mathbb{R}^{3\times3}$ is the moment-of-inertia matrix. The output-matrix $C = [I_4, 0] \in \mathbb{R}^{4\times7}$ extracts the orientation $q = Cx \in \bar{\mathbb{H}}$.

We consider the spacecraft dynamics (1) in closed-loop with a standard quaternion attitude controller [16]

$$\tau(t) = \omega(t) \times J\omega(t) - K_p \vec{e}(t) - K_d \omega(t) \tag{1c}$$

where the error quaternion $e(t) = \pm q(t) \otimes \bar{r} \in \bar{\mathbb{H}}$ between the actual $q(t) \in \bar{\mathbb{H}}$ and desired $r \in \bar{\mathbb{H}}$ orientations of the spacecraft is chosen such that $e_0 \geq 0$ where $\bar{\mathbb{H}}$ is a double-cover of $\mathbb{SO}(3)$. The proportional and derivative gains of the controller are $K_p \in \mathbb{R}^{3\times3}$ and $K_d \in \mathbb{R}^{3\times3}$, respectively. The torque allocation algorithm used to compute actuators commands to supply the desired torque $\tau(t)$ is outside the scope of this paper.

The asymptotic stability of the equilibrium $(e, \omega) = (0, 0)$ for the closed-loop system (1) is certified by the following Lyapunov function [16]

$$V(e, \omega) = 2 - 2e_0 + \omega^\top K_p^{-1} J \omega \tag{2}$$

where $e_0 = |q^\top r| \geq 0$. Although it is not immediately obvious, this Lyapunov function is indeed a positive definition function of the full state $(e, \omega) \in \mathbb{SE}(3)$ [6]. We will use the level-sets of the Lyapunov function (2) as PI sets for the closed-loop system (1)

$$\mathcal{O}(r, \ell) = \left\{ \begin{bmatrix} q \\ \omega \end{bmatrix} : 2 - 2|q^\top r| + \omega^\top K_p^{-1} J \omega \leq 2 - 2\ell \right\}. \tag{3}$$

The level $\ell \in [0, 1]$ will be chosen to ensure that the PI is constraint admissible. The parameterization $2 - 2\ell$ will simplify evaluating whether an equilibrium $(q, 0)$ is contained in the PI set i.e. $e_0 = |q^\top r| \geq \ell$.

B. Constraints

The closed-loop spacecraft dynamics (1b) are subject to state and input constraints. The spacecraft orientation $q(t)$ must be kept out $q(t) \notin \mathcal{K} \subseteq \bar{\mathbb{H}}$ of the cone

$$\mathcal{K}(d, b, \alpha) = \left\{ q \in \bar{\mathbb{H}} : d^\top R(q) b \geq \cos\alpha \right\} \tag{4}$$

where $\alpha \in [0, \pi]$ is the cone angle, $d \in \mathbb{S}^2$ and $b \in \mathbb{S}^2$ are unit-vectors in the inertial and body frames, respectively, and $R(q)$ is the rotation-matrix between these frames, which depends on the spacecraft orientation $q \in \bar{\mathbb{H}}$. The constraint $q(t) \notin \mathcal{K}$ requires that the angle $\cos^{-1}(d^\top R(q) b)$ between the inertial-frame d and body-frame b unit-vectors is sufficiently large $\cos^{-1}(d^\top R(q) b) > \alpha$. Keep-out constraints can be used to e.g. prevent rocket thrust from damaging a nearby object.

This paper exploits an equivalent parameterization of the keep-out cone (4) from [6]

$$\mathcal{K}(d, b, \alpha) = \left\{ q \in \bar{\mathbb{H}} : q^\top P q \geq \cos\alpha \right\} \tag{5}$$

where

$$P = \begin{bmatrix} d^\top b & -(d \times b)^\top \\ -d \times b & db^\top + bd^\top - d^\top b I \end{bmatrix} \in \mathbb{R}^{4\times4}. \tag{6}$$

According to Proposition 2 from [6], the matrix above has eigenvalues $\lambda_+ = +1$ and $\lambda_- = -1$ both with multiplicity 2 and respective eigenvectors

$$U_+ = \frac{1}{\sqrt{2 + 2d^\top b}} \begin{bmatrix} 0 & 1 + d^\top b \\ d + b & -d \times b \end{bmatrix} \in \mathbb{R}^{4\times2} \tag{7a}$$
$$U_- = \frac{1}{\sqrt{2 - 2d^\top b}} \begin{bmatrix} 0 & 1 - d^\top b \\ d - b & d \times b \end{bmatrix} \in \mathbb{R}^{4\times2}. \tag{7b}$$

Our attitude motion planning algorithm will exploit this eigen-structure.

Another common type of state-constraints are keep-in constraints, in which the spacecraft orientation $q(t)$ is kept inside $q(t) \in \mathcal{Q}$ a cone

$$\mathcal{Q} = \left\{ q \in \bar{\mathbb{H}} : q^\top P q \geq \cos\alpha_{\mathcal{Q}} \right\} \tag{8}$$

where P has the form (6). The keep-in cone requires that a body-frame vector $b \in \mathbb{S}^2$ is kept aligned with an inertia-frame vector $d \in \mathbb{S}^2$ within an angle $\alpha \in [0, \pi]$. Keep-in constraints can be used to ensure that the solar panels receive sufficient radiation to power the spacecraft. Keep-in constraints are equivalent to enforcing keep-out constraints $q(t) \notin \bar{\mathbb{H}} \setminus \mathcal{Q}$ on the complement $\mathcal{K} = \bar{\mathbb{H}} \setminus \mathcal{Q}$ of the keep-in set $\mathcal{Q}$.

Bounds on the control torque τ and angular velocity ω can be enforced by limiting the size $\ell \geq \underline{\ell}$ of the PI sets $\mathcal{O}$ where the bound $\underline{\ell}$ can be computed offline [8].

C. Attitude Planning Problem and Algorithm

The attitude planning problem is summarized below.

Problem 1: Find a feasible torque trajectory $\tau(t) \in \mathcal{T}$ such that the spacecraft attitude converges to the target orientation $q(t) \to r_\infty$ as $t \to \infty$ while avoiding keep-out constraints $q(t) \notin \mathcal{K}_k$ and maintaining keep-in constraints $q(t) \in \mathcal{Q}$.

The attitude planning problem can be solved using the ISMP described by Alg 1. The ISMP searches an appropriately constructed directed *search-tree* $\mathfrak{T}$ for a sequence $\{\bar{r}_i\}_{i=1}^N$ of intermediate references $\bar{r}_i \in \bar{\mathbb{H}}$ that guide the spacecraft (1) state $x(t) = (q(t), \omega(t)) \in \mathbb{SE}(3)$ from an initial state $x(0) = x_0$ to a target equilibrium orientation $r_\infty \in \bar{\mathbb{H}}$ while enforcing keep-out constraints $q(t) = Cx(t) \notin \mathcal{K}_k$. The defining feature of the ISMP is that knowledge of the closed-loop spacecraft dynamics (1) is incorporated into the search-tree $\mathfrak{T}$ using its CAPI sets. Associated with each node $i \in \mathfrak{I}$ is a CAPI set $\mathcal{O}_i$, which is both constraint admissible $C\mathcal{O}_i \cap \mathcal{K}_k = \varnothing$ and positive invariant $x(0) \in \mathcal{O}_i \Rightarrow x(t) \in \mathcal{O}_i \ \forall t > 0$. The edges $(i,j) \in \mathfrak{E}$ of the tree $\mathfrak{T} = (\mathfrak{I}, \mathfrak{E})$ indicate that the state (1) will enter the j-th safe-set $\mathcal{O}_j$ while tracking the i-th node without leaving the current safe-set $\mathcal{O}_i$. Thus, the ISMP avoids obstacles by moving the spacecraft state through a sequence of safe-sets $\mathcal{O}_{\sigma_i}$ for $\{\sigma_i\}_{i=0}^N$.

Algorithm 1 Invariant-Set Motion-Planner

1: Search tree $\mathfrak{T}$ for path $\{r_{\sigma_0}, \ldots, r_{\sigma_N}\}$
2: **repeat**
3: **if** $x(t) \in \mathcal{O}_{\sigma_{k+1}}$ **then**
4: $k \leftarrow k+1$
5: **end if**
6: Track current target state $r(t) = r_{\sigma_k}$
7: **until** $r(t) = r_\infty$

In previous work [6], we applied the ISMP Alg 1 to spacecraft attitude motion planning using a pre-built search-graph $\mathfrak{G}$. This paper will focus on the rapid online construction of an appropriate search-tree $\mathfrak{T} = (\mathfrak{I}, \mathfrak{E})$ for Alg 1.

III. Invariant-Set Motion-Planner

Alg 2 describes the construction of the search-tree $\mathfrak{T}$ for the rapidly-exploring variant of the ISMP in Alg 1. Alg 2 incorporates CAPI sets into the RRT algorithm [12], [13]. Alg 2 randomly samples an orientation $r \in \bar{\mathbb{H}}$ which is moved to r_i to form a connection $(i,j) \in \mathfrak{E}$ with the nearest orientation r_j in the search-tree $\mathfrak{T}$. Then a CAPI set $\mathcal{O}_i$ is constructed around this new node $r_i \in \bar{\mathbb{H}}$. Alg 2 continues until it finds a CAPI set $\mathcal{O}_i$ that contains the current state $(q(0), \omega(0)) \in \mathcal{O}_i$ of the spacecraft (1). This section provides four mathematical results that describe the sub-routines for lines 3,6-9 of Alg 2. Note that lines 3 and 9 use the same sub-routine.

A. Sampling Safe Orientations

In this section, we describe line 6 of Alg 2 where we randomly sample orientations $r \in \bar{\mathbb{H}}$. We can sample orientations from the group $\bar{\mathbb{H}} = (\mathbb{S}^3, \otimes)$ by sampling from the

Algorithm 2 Search Tree $\mathfrak{T}$ Construction

1: **Input:** Current state $(q(0), \omega(0))$, target r_∞, obstacles $\mathcal{K}_k$
2: **Output:** Search Tree $\mathfrak{T}$
3: Scale ℓ_∞ PI set $\mathcal{O}_\infty$ for safety $C\mathcal{O}_\infty \cap \mathcal{K}_k = \varnothing, \ \forall k$
4: Initialize tree $\mathfrak{T}$.`add-node` $= (r_\infty, \ell_\infty)$
5: **repeat**
6: Sample random orientation $r \in \mathcal{Q}$
7: Find nearest orientation $r_j \in \mathfrak{T}$ to r
8: Move $r \to r_i \in \text{int}(C\mathcal{O}_j)$ to form edge from r_i to r_j
9: Scale ℓ_i PI set $\mathcal{O}_i$ for safety $C\mathcal{O}_i \cap \mathcal{K}_k = \varnothing, \ \forall k$
10: Update search tree
$$\mathfrak{T}.\texttt{add-node} = \{r_i, \ell_i\}$$
$$\mathfrak{T}.\texttt{add-edge} = (i,j)$$
11: **until** $(q(0), \omega(0)) \in \mathcal{O}_i$

underlying set $\mathbb{S}^3 \subseteq \mathbb{R}^4$ and rejecting samples that violate the keep-out $r \in \mathcal{K}$ and keep-in $r \notin \mathcal{Q}$ constraints. However, this can be inefficient for attitude motion planning problems with restrictive keep-in constraints. The following lemma provides a parameterization of orientations $r \in \mathcal{Q} \subseteq \bar{\mathbb{H}}$ that satisfy the keep-in cone (8).

Lemma 1: The keep-in cone (8) can be written as

$$\mathcal{Q} = \left\{ U(q \otimes p) : \theta_+, \theta_- \in \big[-\pi, \pi\big], \rho \in \big[\cos\tfrac{\alpha}{2}, 1\big] \right\} \quad (9)$$

where $U = [U_+, U_-]$ are the eigenvectors (7) of P (6) and

$$q = \begin{bmatrix} \cos\theta_+/2 \\ \sin\theta_+/2 \\ 0 \\ 0 \end{bmatrix} \text{ and } p = \begin{bmatrix} \rho \\ 0 \\ \sqrt{1-\rho^2}\cos\theta_- \\ \sqrt{1-\rho^2}\sin\theta_- \end{bmatrix} \quad (10)$$

Proof: Let $\mathcal{Q}$ denote the original cone (8) and $\mathcal{Q}'$ denote the alternative cone (9). We will prove $\mathcal{Q} = \mathcal{Q}'$. First, consider $r = U(q \otimes p) \in \mathcal{Q}'$. Since U is the eigenvector-matrix of P, we have $r^\top P r = \|r_+\|^2 - \|r_-\|^2$ where $(r_+, r_-) = q \otimes p$ and $r_+, r_- \in \mathbb{R}^2$ multiply the eigenvalues $\lambda_+ = +1$ and $\lambda_- = -1$, respectively. Since the $r \in \bar{\mathbb{H}}$, we have $\|r\|^2 = \|r_+\|^2 + \|r_-\|^2 = 1$. Or equivalently, $\|r_-\|^2 = 1 - \|r_+\|^2$. Thus, $r^\top P r = 2\|r_+\|^2 - 1$. Substituting $r_+ = (\rho\cos\frac{\theta_+}{2}, \rho\sin\frac{\theta_+}{2})$, we obtain

$$\begin{aligned} r^\top P r &= 2\rho^2\cos(\tfrac{\theta_+}{2})^2 + 2\rho^2\sin(\tfrac{\theta_+}{2})^2 - 1 = 2\rho^2 - 1 \\ &\geq 2\cos(\tfrac{\alpha}{2})^2 - 1 = \cos\alpha \end{aligned}$$

where $\cos(\frac{\theta_+}{2})^2 + \sin(\frac{\theta_+}{2})^2 = 1$ and $\cos(\frac{\alpha}{2})^2 = \frac{1}{2} + \frac{1}{2}\cos\alpha$. Since $r^\top P r \geq \cos\alpha$, we have $r \in \mathcal{Q}$. Therefore, $\mathcal{Q}' \subseteq \mathcal{Q}$.

Finally, we prove $\mathcal{Q} \subseteq \mathcal{Q}'$. Consider $r \in \bar{\mathbb{H}} \setminus \mathcal{Q}'$. Then, $\rho < \cos\frac{\alpha}{2}$. Thus, $r^\top P r = 2\rho^2 - 1 < 2\cos(\frac{\alpha}{2})^2 - 1 = \cos\alpha$ i.e $r \notin \mathcal{Q}$. Therefore, $r \notin \mathcal{Q}' \Rightarrow r \notin \mathcal{Q}$ i.e. $\mathcal{Q} \subseteq \mathcal{Q}'$. ■

The following theorem shows how to uniformly sample the the keep-in cone (9) from Lemma 1. This is important since many of the beneficial theoretical properties of RRT depend on uniform sampling [13].

Theorem 1 (Sampling): Let $\theta_+, \theta_- \sim \mathbb{U}([-\pi, +\pi])$ and $\rho \sim \mathbb{U}([\cos\frac{\alpha}{2}, 1])$. Then $U(\mathbf{q} \otimes \mathbf{p}) \sim \mathbb{U}(\mathcal{Q})$ where the random variables $\mathbf{p}$ and $\mathbf{q}$ are defined by (10).

Proof: First, we will derive the probability density functions (PDFs) of $\mathbf{p}$ and $\mathbf{q}$. Since the scalar function $q_x = \sin\theta_+/2$ is monotonic for $\theta_+ \in [-\pi, \pi]$, the PDF of $\mathbf{q}_x$ is

$$f_{\mathbf{q}_x}(q_x) = f_{\theta_+}(2\sin^{-1} q_x)\frac{d}{dq_x}2\sin^{-1} q_x \propto \frac{1}{\sqrt{1-q_x^2}}$$

where f_{θ_+} is constant since θ_+ is uniformly distributed. We can lift this PDF $f_{\mathbf{q}_x}$ into a PDF $f_{\mathbf{q}}$ for the full quaternion $\mathbf{q} \in \bar{\mathbb{H}}$ as

$$f_{\mathbf{q}}(q) = \delta(q_0 \pm \sqrt{1-q_x^2}) f_{\mathbf{q}_x}(q_x)\delta(q_y)\delta(q_z)$$

where δ is the Dirac delta function. Using the identity $\delta(q^\top q - 1) = 2\sqrt{1-q_x^2}\delta(q_0 - \pm\sqrt{1-q_x^2})$ [14], we have $f_{\mathbf{q}}(q) \propto \delta(q_y)\delta(q_z)\delta(q^\top q - 1)$ i.e. $\mathbf{q}$ is uniformly distributed.

Next, we derive the PDF for $\mathbf{p}$. Restricting (10) to $p_- = (p_y, p_z)$ produces a bijective map $p_- = H(\rho, \theta_-)$. The PDF of $\mathbf{p}_-$ is given by

$$f_{\mathbf{p}_-}(p_-) = f_{\rho,\theta_-}(H^{-1})|\det \nabla H^{-1}| \propto |\det \nabla H^{-1}|$$

where $f_{\rho,\theta_-}(H^{-1}(p_-))$ is constant since ρ and θ_- are uniformly distributed. By direct computation, the Jacobian $\nabla H^{-1}(p_-)$ of the inverse $H^{-1}(p_-)$ of (10) is

$$\nabla H^{-1}(p_-) = \begin{bmatrix} \frac{p_y}{\sqrt{1-\|p_-\|^2}} & \frac{p_z}{\sqrt{1-\|p_-\|^2}} \\ \frac{-p_z}{\|p_-\|^2} & \frac{p_y}{\|p_-\|^2} \end{bmatrix}$$

and thus $|\det \nabla H^{-1}| = 1/\sqrt{1-\|p_-\|^2}$. Applying a similar lifting and identity as above, yields the PDF for $\mathbf{p}$

$$f_{\mathbf{p}}(p) = \delta(p_0 \pm \sqrt{1-\|p_-\|^2})|\det \nabla H^{-1}|\delta(p_x) = \delta(p_x).$$

Thus, both $\mathbf{p}$ and $\mathbf{q}$ are uniformly distributed.

Next, we derive the PDF of $\mathbf{r} = \mathbf{q} \otimes \mathbf{p}$. Since $\mathbf{p}$ and $\mathbf{q}$ are independent, the PDF of $\mathbf{r}$ is by definition

$$\begin{aligned} f_{\mathbf{r}}(r) &= \int\int f_{\mathbf{q}}(q) f_{\mathbf{p}}(p)\delta(r - q \otimes p)dpdq \\ &= \int f_{\mathbf{q}}(q) f_{\mathbf{p}}(\bar{q} \otimes r)\left(\int \delta(r - q \otimes p)dp\right)dq \end{aligned} \quad (11)$$

where $f_{\mathbf{p}}(p)\delta(r - q \otimes p) = f_{\mathbf{p}}(\bar{q} \otimes r)\delta(r - q \otimes p)$. Using the scaling and translation properties of δ, the inner-integral is

$$\int \delta(r - q \otimes p)dp = \int \delta(r - Qp)dp = \frac{1}{\det Q}$$

where $Q \in \mathbb{R}^{4\times4}$ is the matrix representation of the linear operator $q\otimes$. Since $\det Q = 1$, the integral (11) simplifies

$$f_{\mathbf{r}}(r) = \int f_{\mathbf{q}}(q) f_{\mathbf{p}}(\bar{q} \otimes r)dq$$

where $f_{\mathbf{q}}(q) = 1$ if and only if $q = (q_0, q_x, 0, 0) \in \bar{\mathbb{H}}$ is a pure x-rotation. Likewise, $f_{\mathbf{p}}(\bar{q} \otimes r) = 1$ if and only if $p = \bar{q} \otimes r \in \bar{\mathbb{H}}$ is a pure yz-rotation i.e. $p_x = r_x q_0 - r_0 q_x = 0$. Thus, the integral simplifies

$$f_{\mathbf{r}}(r) = \int\int \delta(q_0^2 + q_x^2 - 1)\delta(r_x q_0 - r_0 q_x)dq_0 dq_x$$

Since the line $r_x q_0 - r_0 q_x = 0$ pass through the origin, it intersects the circle $q_0^2 + q_x^2 = 1$ at exactly 2 points. Thus, $f_{\mathbf{r}}(r) \propto 2$ is constant and therefore $\mathbf{r}$ is uniformly distributed.

Finally, we note that the orthogonal matrix U will not affect the uniform distribution i.e. $f_{\mathbf{Ur}}(Ur) \propto f_{\mathbf{r}}(r)$. Thus, $U(\mathbf{q} \otimes \mathbf{p}) \sim \mathbb{U}(\mathcal{Q})$ since $U(q \otimes p)$ parameterized $\mathcal{Q}$ by Lemma 1. ■

B. Nearest-Neighbor Orientation

In this section, we describe line 7 of Alg 2 which finds the nearest orientation $r_j \in \mathfrak{T}$ in the search-tree $\mathfrak{T}$ to the uniformly sampled orientation $r \in \bar{\mathbb{H}}$. As argued in [7], the appropriate metric for measuring the *nearest* orientation is the Minkowski function

$$\Phi_{C\mathcal{O}_i}(r) = \inf_\rho \left\{\rho \geq 0 : r \in \rho C\mathcal{O}(r_i, \ell_i)\right\} \quad (12)$$

of the PI set (3). The Minkowski function (12) quantifies the amount $\rho \geq 0$ we need to scale $\rho C\mathcal{O}$ the PI set $\mathcal{O}$ to contain the point $r \in \rho C\mathcal{O}$. Minkowski functions generalize distance e.g. Euclidean distance is a Minkowski function with a unit-ball and 1-norm and ∞-norms are Minkowski functions with a unit hyperoctohedron or hypercube, respectively [17]. The following proposition provides a closed-form for (12).

Proposition 1 (Minkowski Distance): The Minkowski distance (12) between $r \in \bar{\mathbb{H}}$ and $r_i \in \bar{\mathbb{H}}$ is

$$\Phi_{C\mathcal{O}_i}(q) = |r_i^\top r|/\ell_i. \quad (13)$$

Proof: Since the Lyapunov function (2) does not include cross-terms between the attitude error e and angular velocity ω, we have $C\mathcal{O} = \{e \in \bar{\mathbb{H}} : e_0 \geq \ell\}$ where $e_0 = |r_i^\top r|$. Thus, $\rho C\mathcal{O} = \{\rho e : e_0 \geq \ell\} = \{e \in \bar{\mathbb{H}} : e_0 \geq \rho\ell\}$. The infimum $\rho \geq 0$ for which $e = r_i \otimes r \in \rho C\mathcal{O}$ holds is (13). ■

Using Proposition 1, the index $i \in \mathfrak{I}$ of the nearest node r_i is $\arg\min_{i\in\mathfrak{I}} |r_i^\top r|/\ell_i$.

C. Connecting to Nearest-Neighbor

The edges $(i, j) \in \mathfrak{E}$ of the search-tree $\mathfrak{T} = (\mathfrak{I}, \mathfrak{E})$ indicate that the spacecraft can safely transition from tracking reference r_i to r_j. We add a directed edge $(i, j) \in \mathfrak{E}$ when $r_i \in \text{int}(C\mathcal{O}_j)$. In this section, we describe line 8 of Alg 2 in which we move the sampled reference $r \in \bar{\mathbb{H}}$ to form an edge $r \to r_i \in \text{int}(C\mathcal{O}_j)$.

In Cartesian-space, we can form an edge (i, j) by finding a convex-combination such that $r_i = \lambda r + (1-\lambda)r_j \in \text{int}(C\mathcal{O}_j)$ where $\lambda \in [0, 1]$. In quaternion-space, the equivalent operation is the spherical linear interpolation (SLERP)

$$\texttt{SLERP}(r, r_j, \lambda) = \frac{\sin(\varepsilon - \lambda\varepsilon)}{\sin(\varepsilon)}r_j + \frac{\sin(\lambda\varepsilon)}{\sin(\varepsilon)}q \quad (14)$$

where $\varepsilon = \cos^{-1} r_j^\top r$ is the angle between vectors r and r_j and $\lambda \in [0, 1]$. The following proposition uses the SLERP (14) to form an edge $r_i \in \text{int}(C\mathcal{O}_j)$.

Proposition 2 (Connection): An equilibrium state $(r_i, 0) \in \mathbb{SE}(3)$ satisfies $(r_i, 0) \in \text{int}(\mathcal{O}_j)$ if

$$r_i = \cos\tfrac{\theta}{2} r_j + \sqrt{\frac{(\sin\frac{\theta}{2})^2}{1-(r_j^\top r)^2}}\big(I - r_j r_j^\top\big)r \quad (15)$$

where $\theta \in (0, 2\cos^{-1}\ell_j)$.

Proof: An equilibrium state $(r_i, 0) \in \mathbb{SE}(3)$ satisfies $(r_i, 0) \in \text{int}(\mathcal{O}_j)$ if $e_0 = |r_i^\top r_j| \geq \ell_j$. Thus, we need to find $\lambda \in [0, 1]$ such that

$$\texttt{SLERP}(r, r_j, \lambda)^\top r_j = \frac{\sin(\varepsilon - \lambda\varepsilon)}{\sin\varepsilon} + \frac{\sin\lambda\varepsilon}{\sin\varepsilon}\cos\varepsilon = e_0 \geq \ell_j$$

where $r_j^\top r_j = 1$ since $r_j \in \bar{\mathbb{H}}$ and $\cos\varepsilon = r_j^\top r$ by definition. Using the trig-identity $\sin(\varepsilon - \lambda\varepsilon) = \sin\varepsilon\cos\lambda\varepsilon - \cos\varepsilon\sin\lambda\varepsilon$

$$e_0 = \cos\lambda\varepsilon - \frac{\cos\varepsilon\sin\lambda\varepsilon}{\sin\varepsilon} + \frac{\cos\varepsilon\sin\lambda\varepsilon}{\sin\varepsilon}.$$

Simplifying, yields $\lambda = \cos^{-1} e_0 / \cos^{-1} r_j^\top r$. Substituting into the SLERP, we obtain

$$r_i = \texttt{SLERP}(r, r_j, \lambda) = \frac{\sin(\varepsilon - \lambda\varepsilon)}{\sqrt{1 - (r_j^\top r)^2}} r_j + \frac{\sqrt{1 - e_0^2}}{\sqrt{1 - (r_j^\top r)^2}} r$$

where $\sin\lambda\varepsilon = \sqrt{1 - e_0^2}$ since $\cos\lambda\varepsilon = e_0$ and $\sin\varepsilon = \sqrt{1 - (r_j^\top r)^2}$ since $\cos\varepsilon = r_j^\top r$. Using the identity $\sin(\varepsilon - \lambda\varepsilon) = \sin\varepsilon\cos\lambda\varepsilon - \cos\varepsilon\sin\lambda\varepsilon$ and re-arranging terms yields

$$r_i = \texttt{SLERP}(r, r_j, \lambda) = e_0 r_j + \sqrt{\frac{1 - e_0^2}{1 - (r_j^\top r)^2}}(r - r_j r_j^\top r)$$

which produces (15) for $e_0 = \cos\frac{\theta}{2}$ and $\sqrt{1 - e_0^2} = \sin\frac{\theta}{2}$. ■

Proposition 2 provides a closed-form solution for connecting the edge $(i, j) \in \mathfrak{E}$. The tuning-parameter $\theta \in (0, 2\cos^{-1}\ell_j)$ is the desired angle between the quaternions $r_i, r_j \in \bar{\mathbb{H}}$. For $\cos\frac{\theta}{2} \approx \ell_j$, the orientation r_i is near the boundary of $\mathcal{O}_j$. This can reduce the number of samples required to find a path. However, Alg 1 will be slow to switch modes $r_i \to r_j$. For $\cos\frac{\theta}{2} \approx 1$, $r_i \approx r_j$ is near the "center" r_j of $\mathcal{O}_j$ which will produce smoother inputs.

D. Constructing CAPI Set

Level-sets (3) of the Lyapunov function (2) are PI, but not necessarily constraint admissible. In this section, we describe lines 3 and 9 of Alg 2 where we select the level $\ell_i \in (0, 1)$ to ensure constraint admissibility $C\mathcal{O}_i \cap \mathcal{K} = \varnothing$. The following corollary of Theorem 1 from [6] provides necessary and sufficient conditions for admissibility.

Corollary 1 (Admissibility): The PI set (3) is constraint admissible $C\mathcal{O}(r, \ell) \cap \mathcal{K}(d, b, \alpha) = \varnothing$ if and only if

$$\ell \geq \left\|U_+(d, b)^\top r\right\| \cos\tfrac{\alpha}{2} + \left\|U_-(d, b)^\top r\right\| \sin\tfrac{\alpha}{2} \qquad (16)$$

where U_+ and U_- were defined in (7).

Proof: According to Theorem 1 from [6], the PI set (3) is admissible $C\mathcal{O} \cap \mathcal{K} = \varnothing$ if and only if

$$\|U_+ r\| \leq \ell \text{ and} \qquad (17a)$$

$$\ell\|U_+ r\| + \sqrt{1 - \ell^2}\|U_- r\| \leq \cos\tfrac{\alpha}{2}. \qquad (17b)$$

Consider the change-of-variables $\ell = \cos\bar{\theta}/2 \in [0, 1]$ and $\|U_+ r\| = \cos\beta/2 \in [0, 1]$ for $\bar{\theta}, \beta \in [0, \pi]$. Since $\|r\| = \|U_+ r\| + \|U_- r\| = 1$, we have $\|U_- r\| = \sin\beta/2$. Thus, (17a) can be written as $\cos\beta/2 \leq \cos\bar{\theta}/2$. Or equivalently $\bar{\theta} \leq \beta$.

Condition (17b) can be written as $\cos\frac{\bar{\theta}}{2}\cos\frac{\beta}{2} + \sin\frac{\bar{\theta}}{2}\sin\frac{\beta}{2} \leq \cos\frac{\alpha}{2}$. We can simplify this expression using an angle-sum trig identity to obtain $\cos\frac{\bar{\theta}-\beta}{2} \leq \cos\frac{\alpha}{2}$. Or equivalently $|\bar{\theta} - \beta| \geq \alpha$, which can be split into two disjoint inequalities $\theta \geq \beta + \alpha$ or $\theta \leq \beta - \alpha$. Since $\theta \leq \beta$, the first inequality is unsatisfiable. Furthermore, $\bar{\theta} \leq \beta - \alpha$ is a tighter inequality than $\bar{\theta} \leq \beta$. Applying another angle-sum trig identity we obtain

$$\ell = \cos\tfrac{\bar{\theta}}{2} \geq \cos\tfrac{\beta-\alpha}{2} = \cos\tfrac{\beta}{2}\cos\tfrac{\alpha}{2} + \sin\tfrac{\beta}{2}\sin\tfrac{\alpha}{2}.$$

Substituting $\cos\frac{\beta}{2} = \|U_+ r\|$ produces (16). ■

Corollary 1 describe the necessary and sufficient conditions for the PI set (3) to not intersect a keep-out cone $C\mathcal{O} \cap \mathcal{K} = \varnothing$. For multiple keep-out cones $\mathcal{K}_k$, we use the largest level $\ell = \max_k \ell_k$ (larger ℓs produce smaller PI sets). To enforce input or angular velocity constraints, we can set a lower-bound on the level $\ell \geq \underline{\ell}$.

The proof of Corollary 1 provides intuition about how the level ℓ guarantees safety. An equilibrium $(q, 0) \in \mathbb{SE}(3)$ is contained in the PI set (3) if $e_0 = |q^\top r| \geq \ell$ where the scalar-part $e_0 = \cos\theta/2$ of the error quaternion $e = q \otimes \bar{r}$ is related to the angle θ between orientations $q, r \in \bar{\mathbb{H}}$. Thus, the parameterization $\ell = \cos\bar{\theta}/2$ used in the proof places an upper-bound $\theta \leq \bar{\theta}$ on the error angle θ. The angle $\beta = \cos^{-1}\|U_+ r\|$ measures the minimum angle between the reference quaternion $r \in \bar{\mathbb{H}}$ and an orientation q that aligns $d^\top R(q) b = 1$ the inertia-frame $d \in \mathbb{S}^2$ and body-frame $b \in \mathbb{S}^2$ vectors. Thus, the condition $\bar{\theta} = \beta - \alpha$ ensures that no orientation $q \in \mathcal{O}$ in the PI set $\mathcal{O}$ enters the keep-out cone (5).

IV. Simulation Results

In this section, we demonstrate Algs 1 and 2 for the spacecraft attitude motion planning problem. We will compare our result with previous research [6].

In this simulation, the attitude motion planner must guide the spacecraft out of a "maze" of keep-out cones (5). The objective of this simulation is to provide an unrealistically difficult scenario to stress-test the capabilities of the ISMP. The ISMP must carefully manage the momentum of the spacecraft since moving to avoid a keep-out cone can produce overshoot causing the spacecraft to enter a different keep-out cone. The spacecraft starts from rest $\omega(0) = 0$ at the home-position $q(0) = \mathbb{1}$. The target orientation is a 20° rotation about the x-axis $r_\infty = (\cos 10^\circ, 0, \sin 10^\circ, 0) \in \bar{\mathbb{H}}$. The z-axis of the spacecraft must be kept inside a 22° keep-in cone (8). In addition, the z-axis of the spacecraft must avoid 14 keep-out cones (5) randomly placed in the keep-in cones $\mathcal{Q}$ with keep-out angles $\alpha = 4^\circ$. The arrangement of keep-in and keep-out cones is shown in Fig. 1. For clarity of the plots, the z-axis of the spacecraft was used as the body-frame unit vector $b = (0, 0, 1)$ for both the keep-in and keep-out cones allowing us to project the simulation results onto the xy-plane.

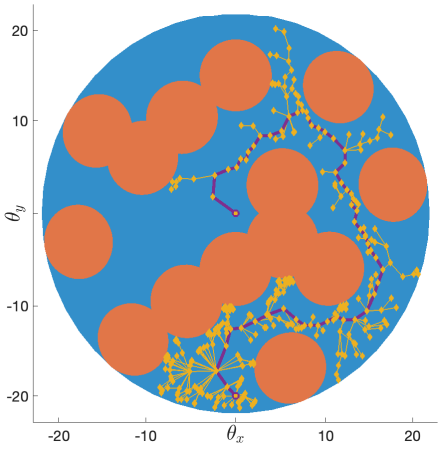

Fig. 1: The blue-circle represents the keep-in cone and the red-circles represent the keep-out cones. The nodes $z \otimes r_i$ for $i \in \mathfrak{I}$ and edges $(z \otimes r_i, z \otimes r_j)$ for $(i,j) \in \mathfrak{E}$ of the search-tree $\mathfrak{T}$ produced by Alg 2 are shown in yellow. The path $z \otimes r_{\sigma_1}, \ldots, z \otimes r_{\sigma_N}$ is shown in purple.

A search-tree $\mathfrak{T} = (\mathfrak{I}, \mathfrak{E})$ was constructed using Alg 2 with $\theta = \cos^{-1} \ell_j$ as shown in Fig 1. The advantage of constructing the search-tree $\mathfrak{T}$ online is that the PI set $\mathcal{O}_i$ are customized for the arrangement of keep-out cones $\mathcal{K}$. In contrast, the previous method [6] used a pre-constructed search-graph $\mathfrak{G} = (\mathfrak{I}_\mathfrak{G}, \mathfrak{E}_\mathfrak{G})$ where nodes $i \in \mathfrak{I}_\mathfrak{G}$ are removed if the pre-scaled PI set $\mathcal{O}_i$ collides with a keep-out cone $C\mathcal{O}_i \cap \mathcal{K} \neq \varnothing$. This requires a dense uniform graph to increase the likelihood of finding a path through any arrangement of constraints. For this example, the previous method [6] used a fixed level $\ell = cos1^\circ$ requiring a search-graph $\mathfrak{G} = (\mathfrak{I}_\mathfrak{G}, \mathfrak{E}_\mathfrak{G})$ with $|\mathfrak{I}_\mathfrak{G}| = 3529$ nodes and $|\mathfrak{E}_\mathfrak{G}| = 119,893$ edges. In contrast, search-tree $\mathfrak{T}$ produced by Alg 2 has $|\mathfrak{I}| = 534$ nodes and $|\mathfrak{E}| = |\mathfrak{I}| - 1$ edges. The customized levels in Alg 2 produced angular-bounds $\bar{\theta}_i$ ranging from 0.06° to 1.9° depending on the proximity to a keep-out cone. This search-tree was constructed by Alg 2 in 1.48 seconds.

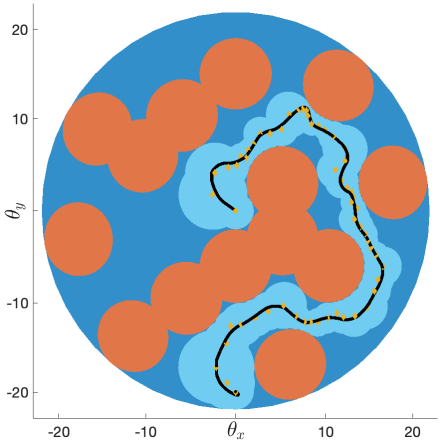

Fig. 2: The path $z(t)$ traced by the z-axis produced by Alg 1 using the search-tree $\mathfrak{T}$ produced by Alg 2.

Alg 1 used the search-tree $\mathfrak{T}$ produced by Alg 2 to find a sequence $\{r_{\sigma_k}\}_{k=1}^{66} \subseteq \mathfrak{I}$ of 66 references $r_{\sigma_k} \in \bar{\mathbb{H}}$ that safely guide the spacecraft from the initial state x_0 to the target equilibrium $(r_\infty, 0)$ as shown in Fig. 1.

Following Alg 1, the reference r_{σ_k} tracked by the closed-loop spacecraft dynamics (1) is updated $k \leftarrow k+1$ each time the state $x(t)$ enters the next CAPI set $\mathcal{O}_{\sigma_{k+1}}$. The switching condition $x(t) \in \mathcal{O}_{\sigma_{k+1}}$ was checked every 1 second. Between updates, the spacecraft dynamics (1) were simulated using MATLAB's `ode45` solver. The path $z(t)$ of the z-axis of the spacecraft is shown in Fig 2. Note that the actual path $z(t)$ of the z-axis of the spacecraft does not perfectly track the planned path $z \otimes r_i$. However, collisions are avoided since $z(t)$ remains in the corridor of CAPI sets $\mathcal{O}_{\sigma_k}$ along the path $\{r_{\sigma_k}\}_{k=1}^{66} \subseteq \mathfrak{I}$ as shown in Fig 2.

REFERENCES

[1] K. Berntorp, A. Weiss, C. Danielson, S. Di Cairano, and I. Kolmanovsky, "Automated driving: Safe motion planning using positive-invariant sets," in *Intelligent Transportation Systems Conference*, 2017.

[2] K. Berntorp, R. Bai, K. F. Erliksson, C. Danielson, A. Weiss, and S. D. Cairano, "Positive invariant sets for safe integrated vehicle motion planning and control," *IEEE Transactions on Intelligent Vehicles*, 2020.

[3] C. Danielson, A. Weiss, K. Berntorp, and S. Di Cairano, "Path planning using positive invariant sets," in *Conf. on Decision and Control*, 2016.

[4] C. Danielson, K. Berntorp, A. Weiss, and S. Di Cairano, "Robust motion-planning for uncertain systems with disturbances using the invariant-set motion-planner," *IEEE Transactions on Automatic Control*, 2019.

[5] C. Danielson, K. Berntorp, S. Di Cairano, and A. Weiss, "Motion-planning for unicycles using the invariant-set motion-planner," in *American Control Conference*, 2020.

[6] C. Danielson, J. Kloeppel, and C. Petersen, "Spacecraft attitude control using the invariant-set motion-planner," *IEEE Control Systems Letters*, pp. 1–1, 2021.

[7] A. Weiss, C. Danielson, K. Berntorp, I. Kolmanovsky, and S. Di Cairano, "Motion planning with invariant set trees," in *Conf. on Control Technology and Applications*, 2017.

[8] A. Weiss, C. Petersen, M. Baldwin, R. Erwin, and I. Kolmanovsky, "Safe positively invariant sets for spacecraft obstacle avoidance," *J. of Guidance, Control, and Dynamics*, 2015.

[9] A. Weiss, F. Leve, M. Baldwin, J. R. Forbes, and I. Kolmanovsky, "Spacecraft constrained attitude control using positively invariant constraint admissible sets on so(3) x r3," in *2014 American Control Conference*, 2014, pp. 4955–4960.

[10] C. Danielson, "Invariant configuration-space bubbles for revolute serial-chain robots," *IEEE Control Systems Letters*, pp. 1–1, 2022.

[11] J.-P. Aubin, *Viability Theory*. Birkhauser Boston Inc., 1991.

[12] S. LaValle and J. Kuffner, "Randomized kinodynamic planning," *The International Journal of Robotics Research*, 2001.

[13] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int J of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[14] M. D. Shuster, "Uniform attitude probability distributions," *The Journal of the Astronautical Sciences*, vol. 51, no. 4, pp. 451–475, 2003.

[15] R. Schneider, G. Rota, B. Doran, P. Flajolet, M. Ismail, T. Lam, and E. Lutwak, *Convex Bodies: The Brunn-Minkowski Theory*. Cambridge University Press, 1993.

[16] B. Wie, H. Weiss, and A. Arapostathis, "Quarternion feedback regulator for spacecraft eigenaxis rotations," *J. of Guidance, Control, and Dynamics*, vol. 12, no. 3, pp. 375–380, 1989.

[17] C. Danielson, "Terminal-cost design for model predictive control with linear stage-costs: A set-theoretic method," *Optimal Control Applications and Methods*, vol. 42, no. 4, pp. 943–964, 2021.