Path Planning of Spatial Rigid Motion with Constrained Attitude

Ran Dai* and Chuangchuang Sun†

This paper presents a general quadratic optimization methodology for autonomous path planning of spatial rigid motion with constrained attitude. The motion to be planned has six degrees of freedom and is assumed under constant velocity in the body frame. The objective is to determine the motion orientation (or attitude), handled as control variables, along the planned paths. A procedure is discussed to transform the rotational constraints and attitude constraints as quadratic functions in terms of unit quaternions and the path planning problem is reformulated as a general quadratically constrained quadratic programming (QCQP) problem. A semidefinite relaxation method is then applied to obtain a bound on global optimal value of the nonconvex QCQP problem. Subsequently, an iterative rank minimization approach is proposed to find the optimal solution. Application examples of aircraft path planning problems are presented using the proposed method and compared with those obtained from the other method.

*Assistant Professor, Aerospace Engineering Department, Iowa State University, Ames, IA. Email: dairan@iastate.edu
†Research Assistant, Aerospace Engineering Department, Iowa State University, Ames, IA. Email: cc-sun@iastate.edu
I. INTRODUCTION

Spatial rigid motion with both translational and rotational evolutions has been found in a variety of dynamical systems, such as robot arms, spacecraft, bio-mechanical systems, aircraft, and many others. There have been continuous efforts to improve the autonomy of dynamical systems when planning spatial transformations [1–3]. Achieving high level autonomy in optimal motion planning requires real-time calculation to optimize desired system performance. Although extensive research has been conducted in this area, there is still plenty of room to improve calculation performance for optimal planning of spatial rigid motion. In this paper, an iterative optimization method based on semidefinite programming is proposed to solve path planning problems for spatial rigid motion with constrained attitude and constant velocity in body frame, where the attitudes are handled as control variables.

The most commonly recognized obstacle that prevents further improvement is the nonlinearity of rigid motion models. Nonlinearity of expressions describing translational and rotational motion makes the real-time path planning extremely difficult, especially when Euler angles are included in the rotational expression. Quaternions have been introduced as mathematical tools for calculations involving three-dimensional (3D) rotations to avoid singularity and reduce expensive computational load created by Euler angle expressions [4]. As alternative and powerful tools for representing object orientation, quaternions have been acknowledged as playing indispensable role in dynamical systems due to their unambiguous, unencumbered, and computationally-efficient features [5]. For example, a unit quaternion based flight kinematics model has been used to find the optimal path with maximum energy harvesting for a solar powered UAV [6]. Instead of using Euler angles, the unit quaternion not only prevents singularity but also has advantage in computation due to its vector notation [7–9]. By using the unit quaternion based method, the aircraft kinematics and attitude constraints, originally expressed by trigonometric functions in terms of Euler angles, are simplified as quadratic functions over unit quaternions. This paper develops a spatial rigid motion model based on unit quaternions for more general path planning problems.
and reformulate them as general quadratically constrained quadratic programming (QCQP) problems.

Approaches to solve path planning problems can be generalized into three categories, indirect methods, direct methods, and evolutionary algorithms [10]. The indirect method, based on Hamiltonian and Euler Lagrange equations, generally requires a good guess of the initial adjoint variables [11]. However, in most such cases, these adjoint variables have no explicit physical meaning, and this causes difficulty in guessing their initial values within a reasonable scale. When random initial adjoint variables are used, more iterations are generally required and/or convergence to a meaningful solution may not be obtained. Although simplified models can be used when applying the indirect method to obtain an optimal solution, most often they will not yield high precision results. Collocation and nonlinear programming, one of the well-known direct methods, transforms the two-point boundary value problem into a parameter optimization problem by discretizing the trajectory into multiple segments characterized by state and control variables as parameters [12,13]. While efficient for solving a large class of problems in different applications, the convergence properties of these algorithms are determined by the initially guessed values of the unknown variables and global optimality is generally not guaranteed. Furthermore, when Euler angles are included in the rotation matrices, singular matrices maybe generated, making the direct method infeasible. Evolutionary algorithms use a fitness function, i.e., the objective to be optimized, to calculate fitness values of candidate solutions, and these values are handled as standards to evaluate the quality of solutions. New evolved solutions are generated according to the fitness values. Parts of both the evolved solutions and the current solutions are chosen to construct next generation solutions. This loop is continued until a stopping criterion is satisfied. The evolutionary approach is quite time consuming as well [14,15].

After reviewing the literature, it reaches to the conclusion that a new and more efficient approach is required to solve real-time rigid motion path planning problems with guaranteed global optimality and fast convergence. The quaternion based rigid motion models formulated in this paper aim at representing rotational constraints and attitude constraints
as quadratic functions to facilitate optimization related calculations and recast the path planning problems as general QCQP problems. Furthermore, a new iterative approach is proposed to solve the general QCQP problem, where the objective and constraint functions are not necessarily convex.

Efforts on solving nonconvex QCQP problems have been focused on finding the bounds on the optimal values by linear or semidefinite relaxation [16, 17]. Although linearization have been used to approximate the nonconvex quadratic constraints, none of them guarantees the optimality of the solution [18, 19]. Another approach is the branch and bound method which can find the global optimal solution of nonconvex QCQP [20, 21]. However, when size of the problem increases, this method is time consuming. In this paper, based on the semidefinite relaxation, the original QCQP problem is transformed into a semidefinite programming (SDP) problem with rank one constraint on the unknown symmetric matrix. Focus in the next step is to find this rank one matrix by proposing an iterative rank minimization (IRM) approach.

The contribution of this paper is composed of two parts. One of them is modeling the path planning problems of spatial rigid motion with attitude constraints as QCQP problems using quaternion representations. Another contribution is the iterative solution of homogeneous QCQP. The organization of the paper is as follows. The translational and attitude constraints using Euler angles and the corresponding path planning problem formulations are introduced in §II. The new model based unit quaternions and relative QCQP formulation are presented in §III. An iterative algorithm to search for the optimal solution of nonconvex QCQP problem is presented in §IV. Simulation examples demonstrating the feasibility and accuracy of the proposed approaches are detailed in §V. Conclusions with a few remarks are summarized in §VI.
II. Problem Formulation using Euler Angles

A. 3D Rotational Model

For a given velocity vector in the body frame, denoted as \( \mathbf{V}_B = [V_x \, V_y \, V_z]^T \in \mathbb{R}^3 \), its relative velocity vector in the inertial frame, denoted as \( \mathbf{V}_I = [\dot{x} \, \dot{y} \, \dot{z}]^T \in \mathbb{R}^3 \), can be found by a direction cosine matrix corresponding to a body 3-2-1 sequence using Euler angles \((\psi, \theta, \phi)\) and is expressed as

\[
\begin{bmatrix}
\dot{x} \\
\dot{y} \\
\dot{z}
\end{bmatrix} =
\begin{bmatrix}
\cos \theta \cos \psi & -\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi & \sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi \\
\cos \theta \sin \psi & \cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi & -\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi \\
-\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta
\end{bmatrix}
\begin{bmatrix}
V_x \\
V_y \\
V_z
\end{bmatrix},
\]

where \( x, y, \) and \( z \) are coordinates in a fixed reference frame, \( V_x, V_y, \) and \( V_z \) are velocity components along the body frame axis. The Euler angles, two reference frames, and their relationship are shown in Fig. (1), where \( x' \) axis is the projection of \( x_b \) axis on the \( X_I - Y_I \) plane.

![Diagram of Euler angles rotations, body frame, and inertial frame.](image)

**Figure 1.** Three Euler angles rotations, body frame, and inertial frame.

The above rotational conversion model represents the relationship between velocity vectors in two reference frames. When velocity in the body frame is constant, the magnitude of the velocity in the inertial frame is constant as well. However, the velocity components in the inertial frame is changing with the rigid body orientation, which is determined by
the Euler angles in expression (1). The velocity in both frames be the same when the iner-
tial frame overlaps with the body frame. By constraining parameters in specific scenarios,
the general rotational conversion model is utilized to represent some well known kinematics
models. For example, by setting $V_y = V_z = 0$, Eq. (1) can represent a three-dimensional
point-mass aircraft kinematics model that is commonly used for formulating path planning
problems under small angle of attack [22, 23]. The simplified model of Eq. (1) under above
assumptions is listed below and illustrated in Figure 2,

$$\begin{align*}
\dot{x} &= V_x \cos \psi \cos \theta \\
\dot{y} &= V_x \sin \psi \cos \theta \\
\dot{z} &= -V_x \sin \theta,
\end{align*}$$

(2)

Furthermore, by constraining $\theta = 0$ in the flight kinematics model, Eq. (1) can be fur-
ther simplified to represent the unicycle model with three degrees of freedom (DOF). Its
mathematical representation is

![Figure 2. 3D aircraft kinematic model.](image)

Determining the orientation that leads to the desired optimal path performance involves
calculation of trigonometric functions even in the simple unicycle model. When general
rotational model in Eq. (1) is considered in path planning formulation, the problem is
highly nonlinear and will be more complicated when attitude constraints are considered.

B. Attitude Constraints

There are different types of attitude constraints. In this paper, the commonly used and challenging one, field-of-view (FOV) constraint, are considered. For certain types of dynamical systems equipped with optical sensors, the FOV of these sensors is associated with the motion orientation and is constrained according to mission requirement. Examples include maintaining visibility of identified landmarks [24], prevention of exposure to the sun glare [25], and increasing the estimation precision [26]. When FOV constraints are considered in path planning, it makes the already complicated problems more challenging.

Figure 2 demonstrates a FOV example for a sensor installed on the aircraft. Generally, a FOV constraint is composed of the boresight vector $S$ and constrained angle $\beta$. For a specified unit vector $S_A$, if $S_A$ is expected to be within or outside the sight of the sensor, the attitude constraints are expressed as

$$
\cos(\beta) \begin{cases} 
\leq S_A \cdot S, & \text{within the sight;} \\
\geq S_A \cdot S, & \text{outside the sight.}
\end{cases}
$$

For example, to prevent sensor from the sun glare, it requires $\cos(\beta) \geq S_A \cdot S$, where $S_A$ is the unit vector of the sunlight direction in the rotational body frame. It is obvious that the Sun direction $S_{A\text{earth}}$ in the Earth fixed frame can be defined by the Sun elevation angle $e$ and azimuth angle $a$ in the form of

$$
S_{A\text{earth}} = \begin{pmatrix}
\cos e \cos a \\
\cos e \sin a \\
\sin e
\end{pmatrix}.
$$
Vector $S_{A_{\text{earth}}}$ can be converted into a vector in the body frame as

$$S_A = R_1(\phi)R_2(\theta)R_3(\psi)S_{A_{\text{earth}}},$$

where $R_1$, $R_2$, and $R_3$ represent rotation matrices about the first, second, and third axis, respectively.

With the above introduction of the rotational and attitude constraints, the path planning problem, such as maximum horizontal range, maximum altitude, and maximum exposure to specified vector, can be formulated using Euler angles. These path planning problems formulated by nonlinear equations are difficult to solve. On one hand, the indirect method requires deriving the necessary conditions for optimality based on Hamiltonian and Euler Lagrange equations. As described in the introduction part, the difficulty of guessing the initial adjoint variables and the trigonometric equations involved in the problem formulation make the indirect method infeasible. On the other hand, the collocation and nonlinear programming method cannot guarantee fast convergence of a global optimal solution when highly nonlinear equations are included in the constraints and/or an initial guess of the solution is randomly selected. Therefore, the unit quaternion based method is introduced to reformulate the path planning problems in a different approach.

III. Problem Formulation based on Unit Quaternion

A. 3D Rotational Model based on Unit Quaternion

An alternative parametrization to describe the rotational motion instead of Euler angles is found by using the unit quaternion. Unit quaternions are generally represented by a $4 \times 1$ vector, where their 2-norm is restricted to be one,

$$\mathcal{U}_q = \{ q \in \mathbb{R}^4 \mid \| q \|_2 = 1 \}. $$
Although the unit quaternion requires one additional variable, they have advantages over the Euler angles, e.g., no singularities and easiness of computation. By Euler’s rotation theorem, a unit quaternion can be visualized as one rotation around an eigen-axis as

\[ q = \begin{bmatrix} \hat{n} \sin \frac{\Theta}{2} \\ \cos \frac{\Theta}{2} \end{bmatrix} = \begin{bmatrix} q_v^T \\ q_0 \end{bmatrix}^T, \]

where \( \hat{n} \) denotes a \( 3 \times 1 \) normalized eigen-axis vector and \( \Theta \) denotes a rotation angle around it. In subsequent sections, the product “\( \otimes \)” refers to quaternion multiplication defined by

\[ q \otimes p = \begin{bmatrix} q_0 p_v + p_0 q_v + q_v \times p_v \\ q_0 p_0 - q_v^T p_v \end{bmatrix}, \]

where \( q = [q_v^T \ q_0]^T \) and \( p = [p_v^T \ p_0]^T \) denote two independent unit quaternions. Another quaternion operation is the “quaternion conjugate” defined as \( q^* = [-q_v^T \ q_0]^T \).

The orthogonal matrix corresponding to a clockwise rotation defined by the unit quaternion is expressed by a homogeneous \( 3 \times 3 \) matrix below,

\[
\begin{bmatrix}
q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_2 + q_1 q_3) \\
2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\
2(q_1 q_3 - q_0 q_2) & 2(q_0 q_1 + q_2 q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2
\end{bmatrix}
\]

Meanwhile, a direction cosine matrix for a 3-2-1 sequence rotation with Euler angles \( \psi, \theta, \phi \) is given in Eq. (1). After comparing the two rotational matrices represented by unit quaternion and Euler angles, the rotational model based on unit quaternion can be rewritten as quadratic
functions in terms of unit quaternions,

\[
\dot{x} = \cos \theta \cos \psi V_x - (\cos \phi \sin \psi - \sin \phi \sin \theta \cos \psi)V_y + (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi)V_z \\
= q^T K_1 q
\]

(9)

\[
\dot{y} = \cos \theta \cos \psi V_x - (\cos \phi \sin \psi - \sin \phi \sin \theta \cos \psi)V_y + (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi)V_z \\
= q^T K_2 q
\]

(10)

\[
\dot{z} = \cos \theta \sin \psi V_x + (\cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi)V_y - (\sin \phi \cos \psi - \cos \phi \sin \theta \sin \psi)V_z \\
= q^T K_3 q
\]

(11)

where \( q^T q = 1 \), \( K_1 = \)

\[
\begin{bmatrix}
V_x & V_y & V_z & 0 \\
V_y & -V_x & 0 & V_z \\
V_z & 0 & -V_x & -V_y \\
0 & V_z & -V_y & V_x \\
\end{bmatrix}
\]

, \( K_2 = \)

\[
\begin{bmatrix}
-V_y & V_x & 0 & -V_z \\
V_x & V_y & V_z & 0 \\
0 & V_z & -V_y & V_x \\
-V_z & 0 & V_x & V_y \\
\end{bmatrix}
\]

, and \( K_3 = \)

\[
\begin{bmatrix}
-V_z & 0 & V_x & V_y \\
0 & V_z & V_y & -V_x \\
V_x & V_y & V_z & 0 \\
V_y & -V_x & 0 & V_z \\
\end{bmatrix}
\]

. For a given velocity in the body frame, \( K_1 \), \( K_2 \), and \( K_3 \) are pre-defined symmetric matrices.

**B. Attitude Constraints based on Unit Quaternion**

In order to obtain the FOV constraints in terms of unit quaternions, a rotation operator using unit quaternions is firstly introduced here. Given a vector \( S \), a rotation of \( S \) around the eigen-axis \( \hat{n} \) is given as

\[
S' = q \otimes S \otimes q^*.
\]

(12)
Since $S$ represents the boresight vector of the sensor, the angle $\varphi$ between the unit vector $S$ and the specified vector $S_A$ can be expressed by the vector dot product

$$\cos(\varphi) = S_A \cdot S', \quad (13)$$

where $S' = q \otimes S \otimes q^*$ denotes the new boresight vector of the sensor when the attitude of the aircraft is determined by $q$. Equation (13) along with this unit quaternion representation enable us to reformulate the FOV constraints in the form of a quadratic function.

**Proposition III.1** The expression of $\cos(\varphi)$ is a quadratic function in terms of unit quaternion $q$.

**Proof** The rotation of $S$ into $S'$ can be rewritten as

$$q \otimes S \otimes q^* = (q_0^2 - \|q_v\|^2) S + 2q_v (q_v \cdot S) + 2q_0 (q_v \times S). \quad (14)$$

It follows that

$$\cos(\varphi) = S_A \cdot S' = (q_0^2 - \|q_v\|^2) S_A \cdot S + 2S_A \cdot q_v (q_v \cdot S) + 2q_0 S_A \cdot (q_v \times S). \quad (15)$$

Since the combination of dot product and cross product yields $S_A \cdot q_v \times S = q_v \cdot S \times S_A$, Eq. (15) can be rewritten as

$$\cos(\varphi) = (q_0^2 - \|q_v\|^2) S_A \cdot S + 2q_v \cdot (S \cdot S_A) \cdot q_v + 2q_0 (q_v \cdot S \times S_A). \quad (16)$$

To reformulate the constraint as a quadratic function, the above function is reorganized by applying the matrices multiplication,

$$\cos(\varphi) = q_0 S_A^T S q_0 - q_v S_A^T S q_v + q_v \cdot (S S_A^T) \cdot q_v + q_v \cdot (S A S^T) \cdot q_v + 2q_0 q_v \cdot S \times S_A$$

$$= q^T M q, \quad (17)$$
where
\[
M = \begin{bmatrix}
S_A S^T + SS_A^T - (S_A^T S) I_3 & S \times S_A \\
S \times S_A^T & S_A^T S
\end{bmatrix}.
\] (18)

and \( q = [q_v^T, q_0]^T \).

By using the quadratic function, the FOV constraints will be expressed as
\[
\cos(\beta) \begin{cases} 
\leq q^T M q, & \text{within the sight;} \\
\geq q^T M q, & \text{outside the sight.}
\end{cases}
\] (19)

With the quaternion representation rotational and attitude constraints model, the two types of path planning problems are reformulated in the following.

C. Path Planning for Maximum Horizontal Range/Altitude using Unit Quaternion

In light of the above discussion on the kinematic model and FOV constraints based on unit quaternions, the optimal path planning problem for maximum range is reformulated as

\[
\max_q \int_{t_0}^{t_f} q^T K_1 q dt
\] (20)

\[
s.t. \quad \dot{y} = q^T K_2 q \\
\dot{z} = q^T K_3 q \\
q^T q = 1
\]

\[
[x(t_0), y(t_0), z(t_0)]^T = [x_0, y_0, z_0]^T
\]

\[
[y(t_f), z(t_f)]^T = [y_f, z_f]^T
\]

\[
\cos(\beta) \leq q^T M q.
\]

Similarly, the optimal path planning problem with FOV constraints for maximum altitude can be reformulated based on unit quaternion by switching \( z \) and \( x \) in the objective and
state constraints, respectively.

Instead of calculating the Euler angles to search for the optimal path, the new formulation
in the above two types of path planning problems will examine the corresponding unit
quaternions which can be handled as the new control variables to optimize the same objective
function.

IV. Solution for Nonconvex QCQP Problem

A. Discretization

The first step to solve the above path planning problems is to discretize the continuous flight
path into a series of segments represented by \( q_h, h = 1, \ldots, H \), at each node, where \( H \) is
the number of discrete nodes. By discretization the coordinate of the aircraft can be integrated
from the initial point to the final point via the discrete nodes, for example,

\[
x_f = x_0 + \sum_{h=1}^{H} q_h^T K_1 q_h \Delta t
\]  

where, \( \Delta t \) is the time step between two adjacent nodes and it is assumed to be uniform in
the current discretization method. Then Eq. (21) can be reformulated as

\[
x_f = x_0 + x^T Q x_i
\]

where \( x = [q_1^T, \ldots, q_H^T]^T \) and \( Q_x \) is denoted as a diagonal matrix with diagonal elements
\( \Delta t K_1 \). For a given \( x_0 \) and a parameter matrix \( Q_x \), finding unknown vector \( x \) to maximize \( x_f \)
is to maximize the quadratic function \( x^T Q_x x \). Under the same setting, the other constraints
and objective function can be approximately expressed as quadratic function of vector \( x \) as
well. For example, the path planning problem for maximum range with FOV constraints
can be reformulated using the discrete nodes $q_h$ below

$$\max_x x^T Q_x x$$

s.t. $y_f = y_0 + \sum_{h=1}^{H} q_h^T K_2 q_h \Delta t = y_0 + x^T Q_y x$

$z_f = z_0 + \sum_{h=1}^{H} q_h^T K_3 q_h \Delta t = z_0 + x^T Q_z x$

$q_h^T q_h = 1, \; h = 1, \ldots, H$

$\cos(\beta) \leq q_h^T M q_h, \; h = 1, \ldots, H,$

where $Q_y$ and $Q_z$ are diagonal matrices with diagonal elements set as $\Delta t K_2$ and $\Delta t K_3$, respectively. With similar modification, the utility constraints and attitude constraints on $q_h$ can be transformed into quadratic equality or inequality constraints of $x$ as well.

From the above analysis, the path planning problems can be generalized as a nonconvex homogeneous QCQP problem in the form of

$$J = \min_x x^T Q_0 x$$

s.t. $x^T Q_j x \leq c_j, \; \forall \; j = 1, \ldots, m$

where $Q_j \in \mathbb{R}^{n \times n} (j = 0, \ldots, m)$ is an arbitrary symmetric matrix and $c_j \in \mathbb{R} (j = 0, \ldots, m)$. For a maximization problem, the objective function is represented by $J = \min_x -x^T Q_0 x$.

Since $Q_j (j = 0, \ldots, m)$ is not necessarily a positive definite matrix, problem in (24) is classified as NP-hard, requiring a global optimization approach for its solution.

QCQP problems have recently attracted significant interests with a wide variety of applications. Basically, any polynomial problems of optimizing a polynomial objective function while satisfying polynomial inequality constraints can be reformulated as general QCQP problems [27]. Extensive relaxation methods have been investigated to obtain the bounds on the optimal value of QCQP. The linear relaxation approach introduces extra variables to transform the quadratic objective and constraints into bilinear terms, which is followed by the linearization of the bilinears [28]. The final linear formulation reaches a bound on
the QCQP optimal value with fast convergence, but low accuracy. The SDP relaxation introduces a rank one matrix to replace the quadratic objective and constraints with linear matrices equalities/inequalities. In general, the SDP relaxation reaches a tighter bound on the optimal value than one obtained from linear relaxation. A detailed discussion of various relaxation approaches and the comparison of their relative accuracy is provided in [29].

**B. The Lower Bound on the Optimal Value of Nonconvex QCQP**

In order to solve the nonconvex QCQP in (24), the semidefinite relaxation method is introduced here to find a tight lower bound on the optimal objective value. By applying interior point method, the relaxed formulation can be solved via a SDP solver [16]. After introducing a rank one matrix $X = xx^T$, the nonconvex QCQP problem in (24) is equivalent to

$$
J = \min X \bullet Q_0 \\
\text{s.t. } X \bullet Q_j \leq c_j, \quad \forall j = 1, \ldots, m \\
X = xx^T,
$$

where `$\bullet$' denotes the trace inner product. However, the rank one constraint $X = xx^T$ is a nonlinear constraint. The last equation in (25) is substituted by a positive semidefinite constraint such that $X \succeq 0$. The semidefinite constraint relaxes the original formulation in (24), which generally yields a tighter lower bound on the optimal value of (24) than the one obtained from linearization relaxation technique [29]. Therefore, by reformulating the problem of (24) in the form of (25), it is feasible to find the lower bound on the optimal value of (24). However, the relaxation method will not yield optimal solution of the unknown variables $x$. Compared to the equivalent transformation in (25), the only difference of the relaxation approach is that the rank one constraint on matrix $X$ is excluded. In order to obtain the optimal solution of $x$, the rank one constraint on matrix $X$ is reconsidered and an IRM approach is proposed to gradually reach the constraint.
C. Iterative Rank Minimization Approach

The heuristic search has been used to minimize the rank of a symmetric or asymmetric matrix over a convex set [30, 31]. Although these methods have successfully lowered the rank of the concerned matrix to certain level, they cannot guarantee that the rank of the matrix is one. When a matrix rank is one, it indicates that the matrix has only one nonzero eigenvalue. Therefore, instead of making constraint on the rank, the focus is on constraining the eigenvalues of $X$ such that the $n-1$ smallest eigenvalues of $X$ are all zero. The eigenvalue constraints on matrices have been used for graph design [32] and are applied here for rank minimization. Before addressing the detailed IRM approach, necessary observations that will be used subsequently in the approach are provided below.

**Proposition IV.1** *The semidefinite constraint*

$$rI_{n-1} - V^T XV \succeq 0$$  \hfill (26)

implies that the second largest eigenvalue $\lambda_{n-1}$ of matrix $X \in \mathbb{R}^{n \times n}$ is less equal than $r$, where $I_{n-1} \in \mathbb{R}^{(n-1) \times (n-1)}$ is the identity matrix, $V \in \mathbb{R}^{n \times (n-1)}$ is full rank matrix whose columns are orthonormal to each other.

**Proof** Equation (26) can be rewritten as

$$V^T (rI_n - X)V \succeq 0,$$

which implies that matrix $rI_n - X$ has $n-1$ nonnegative eigenvalues, where $I_n \in \mathbb{R}^{n \times n}$ is the identity matrix. Assuming the eigenvalues of $X$ is sorted in descending order form, $[\lambda_n, \lambda_{n-1}, \ldots, \lambda_1]$, it follows that $r \geq \lambda_{n-1}$.

**Proposition IV.2** *When $r = 0$ and $X$ is nonzero positive semidefinite matrix, the constraint in (26) implies that $X$ is rank one matrix.*
Proof It’s obvious that when $X$ is a positive semidefinite matrix, $\lambda_n \geq \lambda_{n-1} \geq \ldots \geq \lambda_1 \geq 0$. Since $r \geq \lambda_{n-1}$ and $r = 0$, then $\lambda_n \geq 0 \geq \lambda_{n-1} = \ldots = \lambda_1 = 0$. For nonzero matrix $X$, $\lambda_n > 0$ and it is the only nonzero eigenvalue of $X$. Thus, $X$ is rank one matrix.

However, proposition (IV.1) does not imply that when $r \geq \lambda_{n-1}$, Eq. (26) exists for any $V$ that has orthonormal columns. Based on this fact, it requires finding the necessary conditions for proposition (IV.1).

**Proposition IV.3** The second largest eigenvalue $\lambda_{n-1}$ of matrix $X \in \mathbb{R}^{n \times n}$ is less equal than $r$ if and only if $rI_{n-1} - V^TXV \succeq 0$ where $V \in \mathbb{R}^{n \times (n-1)}$ are the eigenvectors corresponding to the $n-1$ smallest eigenvalues of $X$.

Proof The necessary conditions can be proved by setting the above $V$ as one special case of proposition (IV.1). The sufficiency proof starts from $r \geq \lambda_{n-1}$. In addition,

$$
V^TXV = \begin{bmatrix}
\lambda_{n-1} & 0 & \cdots & 0 \\
0 & \lambda_{n-2} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \lambda_1
\end{bmatrix}
$$

thus, $rI_{n-1} - V^TXV \succeq 0$.

**Corollary IV.4** When $r = 0$ and $X$ is nonzero positive semidefinite matrix, $X$ is rank one if and only if $rI_{n-1} - V^TXV \succeq 0$ where $V \in \mathbb{R}^{n \times (n-1)}$ are the eigenvectors corresponding to the $n-1$ smallest eigenvalues of $X$.

From the above discussion, the rank one constraint in Eq. (25) will be substituted by the semidefinite constraint

$$
 rI_{n-1} - V^TXV \succeq 0 \quad (28)
$$
where \( r = 0 \) and \( \mathbf{V} \in \mathbb{R}^{n \times (n-1)} \) are the eigenvectors corresponding to the \( n-1 \) smallest eigenvalues of \( \mathbf{X} \). However, before solving \( \mathbf{X} \), the exact \( \mathbf{V} \) matrix is not available, thus an iterative method is proposed to gradually minimize the rank of \( \mathbf{X} \). At each step \( k \), the algorithm will solve the following semidefinite programming problem, formulated as

\[
J = \min_{\mathbf{X}_k, r_k} \mathbf{X}_k \cdot \mathbf{Q}_0 + w^k r_k
\]

subject to

\[
\begin{align*}
\mathbf{X}_k \cdot \mathbf{Q}_j &\leq c_j, \; \forall \; j = 1, \ldots, m \\
\mathbf{X}_k &\succeq 0 \\
r_k \mathbf{I}_{n-1} - \mathbf{V}_{k-1}^{T} \mathbf{X}_k \mathbf{V}_{k-1} &\succeq 0,
\end{align*}
\]

where \( w > 1 \) is a weighting factor and \( \mathbf{V}_{k-1} \) are the eigenvectors corresponding to the \( n-1 \) smallest eigenvalues of \( \mathbf{X}_{k-1} \) solved at previous step \( k-1 \). At each step, the algorithm will optimize the original objective function and at the same time minimize parameter \( r \) such that when \( r = 0 \), the rank one constraint on \( \mathbf{X} \) is satisfied. The above approach is repeated until \( r \leq \delta \), where \( \delta \) is a small threshold for stopping criteria. The IRM algorithm can be summarized into three steps:

**Step1** Initialization: Set \( k = 0 \), solve (25) with semidefinite constraint \( \mathbf{X} \succeq \mathbf{x}\mathbf{x}^T \) and obtain \( \mathbf{V}_0 \) from \( \mathbf{X}_0 \), set \( k = k + 1 \);

**Step2** Iteration: while \( r_k > \delta \), solve problem (29) and obtain \( \mathbf{X}_k, r_k \), then update \( \mathbf{V}_k \) from \( \mathbf{X}_k \) and set \( k = k + 1 \);

**Step3** Output: when \( r_k \leq \delta \), find \( \mathbf{x} \) from \( \mathbf{X} \).

In the following, the convergence analysis and optimality conditions are provided in the content of homogenous QCQP formulation. At each IRM step, the subproblem considered for homogeneous QCQP in Eq. (29) can be rewritten with two sets of equality and inequality.
constraints in the form of

\[
J = \min_{X_k, r_k} X_k \cdot Q_0 + w^k r_k \tag{30}
\]

s.t. \quad X_k \cdot Q_i + c_i \leq 0, \ \forall \ i \in \mathcal{I} \nonumber
\]

\[
X_k \cdot Q_j + c_j = 0, \ \forall \ j \in \mathcal{E} \nonumber
\]

\[
r_k I_{n-1} - V_{k-1}^T X_k V_{k-1} \succeq 0 \nonumber
\]

\[
X_k \succeq 0, \nonumber
\]

where \(\mathcal{I}\) and \(\mathcal{E}\) represent the index sets of inequality and equality constraints, respectively.

**Proposition IV.5** \(\lim_{k \to +\infty} r_k = 0\) in the IRM algorithm if \(\exists k > 1\) such that the formulated in (30) is bounded optimization problem.

**Proof** A Lagrange dual function of (30) is constructed as

\[
\mathcal{L} = X_k \cdot Q_0 + w^k r_k + \sum_{i \in \mathcal{I}} \lambda_i (X_k \cdot Q_i + c_i) + \sum_{j \in \mathcal{E}} \mu_j (X_k \cdot Q_j + c_j) \nonumber
\]

\[
- \text{trace}(S_1 (r_k I_{n-1} - V_{k-1}^T X_k V_{k-1})) - \text{trace}(S_2 X_k), \tag{31}
\]

where \(\lambda_i \in \mathbb{R} > 0, \ \mu_i \in \mathbb{R}, \ S_1 \in \mathbb{S}^{n-1}_+\) and \(S_2 \in \mathbb{S}^n_+\) are the Lagrange dual multipliers. The dual function is then expressed as

\[
g(\lambda, \mu, S_1, S_2) = \inf_{X_k, r_k} \mathcal{L}(X_k, r_k, \lambda, \mu, S_1, S_2). \tag{32}
\]

Based on the Lagrange dual in (31), the first order condition is derived as

\[
\frac{\partial \mathcal{L}}{\partial X_k} = Q_0 + \sum_{i \in \mathcal{I}} \lambda_i Q_i + \sum_{j \in \mathcal{E}} \mu_j Q_j + V_{k-1}^T S_1 V_{k-1} - S_2 = 0 \nonumber
\]

\[
\frac{\partial \mathcal{L}}{\partial r_k} = w^k - \text{trace}(S_1) = 0. \tag{33}
\]
Consequently, the dual problem is shown as follows,

\[ J = \max_{\lambda, \mu, S_1, S_2} \sum_{i \in I} \lambda_i c_i + \sum_{j \in E} \mu_j c_j \]

s.t. \[ Q_0 + \sum_{i \in I} \lambda_i Q_i + \sum_{j \in E} \mu_j Q_j + V_{k-1}^T S_1 V_{k-1} - S_2 = 0 \]
\[ w^k - \text{trace}(S_1) = 0 \quad (34) \]
\[ \lambda_i \geq 0, S_1 \succeq 0, S_2 \succeq 0. \]

It is obvious that the problem described in (30) is convex. Moreover, it can be verified that the Slater’s constraints are satisfied. It follows that the strong duality holds.

For bounded optimization problem of (30), the dual problem in (34) is also bounded. Thus \( \lambda, \mu, S_1, S_2 \) are finite variables. As strong duality holds, at the optimal solution point \((X^*_k, r^*_k, \lambda^*, \mu^*, S^*_1, S^*_2)\), it can be shown that

\[ Q_0 \cdot X_k^* + w^k r_k^* = \sum_{i \in I} \lambda_i^* c_i + \sum_{j \in E} \mu_j^* c_j \]

For a weighting factor selected with \( w > 1 \), it can be verified that

\[ \lim_{k \to +\infty} r_k^* = \lim_{k \to +\infty} \frac{\sum_{i \in I} \lambda_i^* c_i + \sum_{j \in E} \mu_j^* c_j - Q_0 \cdot X_k^*}{w^k} = 0. \]

**Proposition IV.6** \( X_k \) converges to the optimal solution \( X^* \) in the IRM when \( \mathcal{E} \neq \emptyset \) and there exists at least one \( c_i \neq 0 \).

**Proof** When \( k \to \infty, r \to 0 \), the third constraint in (30) will become

\[ -V_k^T X_{k+1} V_k \succeq 0 \quad (35) \]

For semidefinite matrix \( X_{k+1} \succeq 0 \), it leads to

\[ V_k^T X_{k+1} V_k = 0. \quad (36) \]
Since the rank of $X_k$ is no more than one, it satisfies

$$V_k^T X_k V_k = 0.$$  \hspace{1cm} (37)

Subtracting (37) from (35) yields

$$V_k^T (X_{k+1} - X_k) V_k = 0.$$  \hspace{1cm} (38)

As $X_k$ is a positive semidefinite matrix with rank one, it follows that

$$X_{k+1} = \alpha X_k.$$  \hspace{1cm} (39)

when $E \neq \emptyset$ and there exists at least one $c_i \neq 0$, $\alpha = 1$ and $\lim_{k \to +\infty} X_k = X^*$. Otherwise, the equality constraint set can not be satisfies in (30). However, when $E = \emptyset$, $X_{k+1} = \alpha X_k$, where $\alpha$ is not necessarily to be equivalent to one.

**V. SIMULATION EXAMPLES**

**A. Example One: Maximum Range**

Two simulation examples for path planning of an aircraft with FOV constraints are illustrated in this section. For the maximum range simulation, the aircraft starts from $[x_0, y_0, z_0] = [0, 0, 0]$ and the final states are set as $[y_f, z_f] = [200, 600]$ meters. The aircraft is flying at constant speed $V_x = 15$ m/s and the flight time is assigned to be $t_f = 300$ seconds. In addition, it is assumed that the aircraft is blocked from the sun glare to protect the optical sensor. The FOV is defined by the constrained angle $\beta = 66^\circ$ and the boresight vector $S$ which is along the body x axis of the aircraft. The Sun is assumed to have constant azimuth angle $a = 45^\circ$ and elevation angle $e = 45^\circ$ during the flight interval. Moreover, the maximum magnitude of change rate of the unit quaternion, denoted as $|\dot{q}_{\text{max}}|$, is set as $0.0022$ and the threshold $\delta$ is set as $10^{-3}$. The corresponding parameters used in the
maximum range simulation are provided in Table 1.

Table 1. Boundary conditions and parameters used in maximum range simulation

| $[x_0, y_0, z_0]$ (m) | $[y_f, z_f]$ (m) | $t_f$ (sec) | $V_x$ (m/s) | $|q_{max}|$ | $a$ (deg) | $e$ (deg) | $\beta$ (deg) | $\delta$ |
|----------------------|------------------|-------------|-------------|-------------|-----------|----------|-------------|---------|
| [0,0,0]              | [200,600]        | 300         | 15          | 0.0022      | 45        | 45       | 66          | $10^{-3}$ |

Figure 3. Simulation result of maximum range flight trajectory from IRM and NLP

Figure 4. Time history of maximum range flight coordinates and Euler angles from IRM and NLP
The path to be optimized is discretized into 10 nodes with a unit quaternion representing its attitude at each node. By implementing the IRM method proposed in §IV, the optimal path is generated in Figure 3 leading to a maximum range of 4.65 km. The corresponding coordinates \([x, y, z]\) derived by integration of the discrete nodes and the Euler angles obtained from the unit quaternions are demonstrated in Figure 4. In order to compare the performance of the new method, comparative simulation results are provided from nonlinear programming (NLP) solver [33] using three groups of random initial guess for the unknown variables. As demonstrated in Figure 3 and Figure 4, the results from the NLP method generate distinct solutions under three groups of random initial guess. Solution 1 from NLP yields a maximum range of 3.44 km, which is not global optimal solution. Solution 3 from NLP is exactly the same as the solution from IRM. Solution 2 from NLP yields the same maximum range of 4.65 km as the IRM solution with different trajectory.

After comparison, it’s obvious that the results from NLP are dependant on the initial guess, which cannot guarantee the global optimal solution. However, the optimal value obtained from the IRM method is consistent with the optimal value bound generated from the relaxed SDP in Eq. (25). In addition, all of the constraints are satisfied in the solution. It can be claimed that the IRM method generates a global optimal solution in this simulation example. Further analysis demonstrates the second largest eigenvalue of matrix \(X\) in each iterative step in Figure 5. It’s shown that \(\lambda_{n-1}\), which is represented by \(r\) in Eq. (29), quickly reduces to zero within 6 steps. Figure 5 indicates that a rank one matrix of \(X\) can be obtained within a few iterative steps.

B. Example Two: Maximum Altitude

For the maximum altitude simulation, the boundary conditions and corresponding parameters used in the simulation are provided in Table 2. The FOV constraint in this case requires the specified vector \(S_A\) which has azimuth \(a = 45^\circ\) and elevation \(e = 30^\circ\) to stay within the view zone of the sensor.

The generated 3D trajectories and the corresponding time history of coordinates and
Figure 5. The second largest eigenvalue \( \lambda_{n-1} \) of matrix \( X \) at each iterative step

Table 2. Boundary conditions and parameters used in maximum altitude simulation

<table>
<thead>
<tr>
<th>( [x_0, y_0, z_0] ) (m)</th>
<th>( [x_f, y_f] ) (m)</th>
<th>( t_f ) (sec)</th>
<th>( V_x ) (m/s)</th>
<th>( \dot{q}_{\max} )</th>
<th>( a ) (deg)</th>
<th>( e ) (deg)</th>
<th>( \beta ) (deg)</th>
<th>( \delta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>[0,0,0]</td>
<td>[2500,2500]</td>
<td>300</td>
<td>15</td>
<td>0.0033</td>
<td>45</td>
<td>30</td>
<td>60</td>
<td>10^{-3}</td>
</tr>
</tbody>
</table>

Euler angles using IRM and NLP approaches are provided in Figs. 6 and 7, respectively. In the second case, the trajectory of solution 1 from NLP is consistent with the trajectory generated from IRM. Both solution yield maximum altitude of 1.13 km. Solution 2 from NLP yields the same optimal value with different trajectories. Furthermore, the solution of the IRM method converges to the optimal solution within 7 steps, as demonstrated in Figure 8.

C. Discretization and Uncertainties

In this section, the appropriate number of discretization nodes and uncertainties are discussed. In both examples, the calculation time is dependant on number of the discrete nodes, the stopping threshold and the performance of the SDP solver. When 10 nodes are used for the discretization, the problem includes 41 unknown variables in each iterative step. It takes the SDP solver around 3 seconds to calculate the solution at each step on an Lenovo
The T430 laptop with Intel i7 CPU and 8GB RAM. Based on the fact that most of the cases will converge to the optimal solution within 7 iterations, it generally takes less than 30 seconds to obtain an optimal solution. Comparing to the NLP method, less computation time is required to obtain a solution with guaranteed global optimality.

More discretization nodes can be added to the path to refine the solution. Theoretically,
the refined path with more discretization nodes is expected to generate more accurate solution. For example one, 15 nodes are used to recalculate the maximum range path. The solution again converges to rank one constraint within 8 iterations and yields a maximum range of 4.51 km. From comparison, when the number of discretization nodes is increased by 50%, it yields similar results with 3.01% increment of accuracy in terms of the performance index. However, due to the limited performance of existing SDP solver, i.e., SeDuMi [34], even though each iteration is solving a convex optimization problem, the calculation time will increase dramatically when the size of SDP problem increases. Therefore, a good balance between the accuracy of the solution and computation time is required according to the definition of the problem.

Simulation results presented in both cases are open-loop solutions without considering uncertainties. When the aircraft deviates from the planned path or parameters, i.e., final boundary conditions, are updated during the mission, new path planning procedure incorporating the updated states and parameters is required under these scenarios. Depending on the requirements of the fight mission and computational environments, the planned path can be recalculated using the proposed IRM method when updates are incorporated at sampling time. However, due to the limited calculation performance, the time interval between the update is set to be larger than the computational time required for each calculation.

Figure 8. The second largest eigenvalue $\lambda_{n-1}$ of matrix $X$ at each iterative step
VI. CONCLUSIONS

This paper examines certain types of path planning problems for spatial rigid motion with attitude constraints and constant velocity in the body frame. It started with reviewing the path planning problem formulation based on Euler angles and identifying the challenges in finding global optimal solution using existing methods. A new approach based on the unit quaternion is then proposed to reformulate the corresponding path planning problems as general quadratic optimization problems. The problem formulation is simplified in the new modeling system and an iterative rank minimization algorithm is proposed to solve the nonconvex quadratically constrained quadratic programming problem. The efficiency and optimality of the new approach are verified by simulation examples.

References


7 Lee, U. and Mesbahi, M., “Spacecraft reorientation in presence of attitude constraints via logarithmic


19Morgan, D., Chung, S.-J., and Hadaegh, F. Y., “Model Predictive Control of Swarms of Spacecraft


