

A Hybrid ADMM for Six-Degree-of-Freedom Entry Trajectory Optimization Based on Dual Quaternions

CHAOYING PEI 

Purdue University, West Lafayette, IN USA

CHANGHUANG WAN

Tuskegee University, Tuskegee, AL USA

RAN DAI , Member, IEEE

Purdue University, West Lafayette, IN USA

JEREMY R. REA

NASA Johnson Space Center, Houston, TX USA

This article investigates the six-degree-of-freedom (6-DoF) entry trajectory optimization problem in a Human-Mars entry, powered descent, and landing mission. During the entry phase, aerodynamic forces are employed to decelerate the vehicle. Instead of being treated as a point mass, both translational and rotational motions of the entry vehicle are considered. Specifically, the 6-DoF rigid body motion of the entry vehicle is modeled using the unit dual quaternion representations to avoid highly nonlinear terms in the flight dynamics expression originally based on the flight-path coordinates. Then, the entry trajectory optimization problem is to minimize the terminal speed subject to dynamical, operational, and mission constraints modeled by the new representation scheme. By applying the discretization technique and polynomial approximation, the entry trajectory optimization problem is reformulated as a nonconvex quadratically constrained quadratic programming problem, which is solved via a hybrid alternating direction method of multipliers (ADMM). The accuracy of the dual-quaternion-based model and the computational efficiency of the hybrid ADMM are validated via numerical simulations.

Manuscript received 17 January 2022; revised 23 May 2022 and 19 September 2022; accepted 13 November 2022. Date of publication 21 November 2022; date of current version 9 June 2023.

DOI No. 10.1109/TAES.2022.3223333

Refereeing of this contribution was handled by G. Zhang.

Authors' addresses: Chaoying Pei and Ran Dai are with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907 USA, E-mail: (peic@purdue.edu; randai@purdue.edu); Changhuang Wan is with the Department of Aerospace Science Engineering, Tuskegee University, Tuskegee, AL 36088 USA, E-mail: (cwan@tuskegee.edu); Jeremy R. Rea is with Flight Mechanics and Trajectory Design Branch, NASA Johnson Space Center, Houston, TX 77058 USA, E-mail: (jrea@alum.mit.edu). (*Corresponding author: Ran Dai.*)

0018-9251 © 2022 IEEE

I. INTRODUCTION

Entry, powered descent, and landing (EDL) operations in a Mars exploration mission require to guide a spacecraft from the Mars atmosphere interface to a desired landing location with a near-zero speed. In this article, we focus on the six-degree-of-freedom (6-DoF) entry trajectory optimization for a Human-scale entry vehicle to guide the vehicle to a designated terminal altitude with minimum terminal speed. Due to the large variations of Mars atmosphere density [1], the complexity of entry vehicle dynamics, including both translational and rotational motion, and mission and operational constraints, it is computationally intractable to generate optimal control commands considering the translational and rotational motion simultaneously while satisfying all specified constraints [2]. There are extensive studies on optimal entry guidance in interplanetary exploration missions. In general, two types of approaches have been developed for optimal entry guidance [3]. One is reference trajectory-based guidance, where the reference trajectory is generated offline, and then, tracked in real time. For example, a second-order cone programming (SOCP) algorithm has been applied to generate a 3-D optimal entry path [4]. Linear pseudospectral model predictive control has been applied for onboard entry trajectory optimization [5]. On the other hand, with the improvement of high-performance computing technology, numerical predictor-corrector guidance (NPCG) has obtained more attention. Instead of relying on a scheme with separate trajectory planner and tracking controller, NPCG algorithms have the capability to accommodate large trajectory dispersion and are computationally viable for real-time implementation [6], [7], [8]. For example, the authors in [6] has applied the predictor-corrector method to a broad scope of entry vehicles with different lift-to-drag ratios. The authors in [9] compared the performance of the NPCG results using bank angle control and direct force control, where the direct force control method has verified advantages in terms of reducing the open-loop flight errors. Most of the existing works of entry guidance schemes consider the translational trajectory optimization and entry vehicle attitude control as two separate problems [1], [10], [11], [12]. Specifically, works focusing on generating optimal trajectories assume a point-mass vehicle that only considers three-degree-of-freedom (3-DoF) translational dynamics. For example, the authors in [13] have developed a trajectory planner for atmospheric entry to generate a feasible trajectory by controlling the bank angle. With a reference entry trajectory, different controllers have been designed to track the reference path. For example, the authors in [10] and [14] have designed adaptive controllers to reduce tracking errors due to uncertainties during the flight.

However, considering the coupling relationship of the translational and rotational motion, instead of solving two separate problems, the rotational and translational motion can be integrated when searching for the optimal control commands to guide the entry vehicle. Then, the constraints related to the rotational motion can be considered directly in the entry trajectory optimization. 6-DoF dynamical models have been considered in trajectory optimization for manipulators [15], industrial robots [16], and reusable launch

vehicles [17]. In addition, the dual-quaternion-based 6-DoF dynamical model that integrates translational and rotational motion has been applied in the powered descent guidance problem [18]. However, there is no attempt of applying the iterative SOCP to the 6-DoF entry trajectory optimization problem. Existing work related to 6-DoF entry guidance focuses on entry process simulation considering aerodynamic-propulsive interactions [19] or decomposes the problem into trajectory planning and attitude control problems [20]. In this article, we consider the 6-DoF entry trajectory optimization problem using the dual quaternion representations. Furthermore, different from the bank angle or direct force control, the net moment control is introduced to improve the flexibility when searching for the optimal trajectory. Different from the existing literature on entry trajectory optimization that describes the translational motion using flight-path coordinates and the rotational motion using Euler angles [4], [6], [11], [21], the dual-quaternion-based model describes the spatial motion of the entry vehicle in a compact form. Furthermore, since the traditional entry dynamical model using flight-path coordinates involves trigonometric functions, it leads to highly nonlinear terms in the entry trajectory optimization problem and subsequent computational complexity when searching for the optimal solution. Moreover, linearization of highly nonlinear terms in the optimization operation generally leads to reduced precision. The dual-quaternion-based model avoids highly nonlinear terms and singularity when depicting the rotation of a rigid body [22], [23].

By combining the dynamics and multiple constraints based on dual quaternions, the entry trajectory optimization problem is to minimize the terminal speed with a designated terminal altitude while satisfying all specified constraints. The reduced speed at the terminal point of the entry phase will benefit fuel saving in the following powered descent phase. By approximating the nonpolynomial functions, e.g., the air density function, using continuous polynomials, the entry trajectory optimization is reformulated as a polynomial optimal control problem. After applying discretization techniques and introducing new variables, the polynomial optimal control problem can be equivalently expressed as a nonconvex quadratically constrained quadratic programming (QCQP) problem.

QCQP has been applied in a wide range of optimization problems, such as optimal power flow [24], signal processing [25], and sensor network localization [26]. Extensive numerical methods have been developed to solve general/nonconvex QCQPs, which can be categorized into relaxation methods and successive convex approximation (SCA) [27], [28]. The relaxation methods can obtain a lower bound on the cost function, which cannot guarantee to find an optimal or even a feasible solution for nonconvex QCQPs in most cases. On the other hand, the SCA approaches solve QCQPs via sequential convex optimization, where each iteration is solved via a convex solver, e.g., Mosek [29]. When another iterative approach, e.g., interior point method [30], is applied to solve each sequential problem formulated in SCA, the multiloop iterations increase the computational load. The alternating direction method of multipliers (ADMM) is a well-known sequential

algorithm for convex optimization that divides an original problem into subproblems and solves them in an alternating sequence. It aims to reduce the computational complexity by fixing a subset of variables when solving each subproblem. The ADMM [31], [32] and its variants [33], [34] are designed initially for convex optimization problems. Recent work in [35] and [36] has applied the ADMM framework to solve nonconvex QCQPs with proven convergence under mild conditions. However, the entry trajectory optimization problem that includes a long range of altitude and speed profile requires many iterations to achieve high precision when employing the ADMM framework to solve the formulated QCQP problem [37], [38], [39]. To accelerate the convergence rate while providing flexibility to search for an optimal solution, a hybrid ADMM algorithm is proposed. The new approach for solving large-scale QCQPs starts with the traditional ADMM framework until the errors from constraints meet a specified threshold. Different from the customized ADMM introduced in [39], a consensus constraint is introduced into the ADMM procedures, which forces the two sets from ADMM primary updates to equal to their average at each iteration to accelerate the convergence rate with a guaranteed linear convergence rate. The analysis of the bounded error and convergence property of the hybrid ADMM is provided. To verify the accuracy of the dual-quaternion-based entry dynamical model and the efficiency of the hybrid ADMM algorithm, simulation examples with comparative results from the traditional dynamical model and a nonlinear programming (NLP) solver are provided.

The contributions of this article include the following:

- 1) integration of the translational and rotational motion of an entry vehicle in a compact form represented by dual quaternions;
- 2) development of a hybrid ADMM algorithm for solving large-scale QCQPs with a rigorously proved bounded error and linear convergence rate;
- 3) applying the hybrid ADMM algorithm to the atmospheric entry trajectory optimization problem using a dual quaternion formulation.

This article is organized as follows. Section II introduces the dual quaternion and the definition of coordinate frames. Section III describes the formulation of atmospheric entry trajectory optimization problem based on dual quaternion representations and flight-path angle. Section IV presents the conversion of the trajectory optimization problem into a homogeneous QCQP. Section V proposes the hybrid ADMM algorithm and its convergence analysis. Simulation results are provided in Section VI and finally, Section VII concludes this article. All the omitted proofs are presented in Appendix.

II. PRELIMINARY AND NOTATIONS

A. Introduction of Dual Quaternion

A quaternion $\hat{\mathbf{q}}$ is represented by a scalar q_0 and a vector $\mathbf{q} = [q_1, q_2, q_3]^T$, which can be written in a vector form as

$$\hat{\mathbf{q}} = [q_0, q_1, q_2, q_3]^T = [q_0, \mathbf{q}^T]^T \in \mathbb{H} \quad (1)$$

where \mathbb{H} denotes the algebra of quaternions. Unit quaternions with their norm equals to 1 are usually used to describe orientations and spatial rotations of objects in three dimensions [40].

Dual quaternion is an extension of quaternion to efficiently express both rotation and translation between two coordinate frames [41]. A dual quaternion $\tilde{\sigma}$ has the form

$$\tilde{\sigma} = \hat{\mathbf{p}} + \epsilon \hat{\mathbf{q}} \in \mathbb{Q} \quad (2)$$

where ϵ is the dual unit, which have the properties $\epsilon^2 = 0$ and $\epsilon \neq 0$, \mathbb{Q} denotes the algebra of dual quaternions, with $\hat{\mathbf{q}}$ defined in (1), similarly, $\hat{\mathbf{p}} \in \mathbb{H}$ is given as $\hat{\mathbf{p}} = [p_0, p_1, p_2, p_3]^T = [p_0, \mathbf{p}^T]^T$.

The dual-quaternion-related operations can be found in [41]. Among them, the multiplication of a matrix $\mathbf{K} \in \mathbb{R}^{8 \times 8}$ and a dual quaternion $\tilde{\sigma} \in \mathbb{Q}$ is commonly used. To better define the multiplication operation between a matrix and a dual quaternion, a dual quaternion set can be embedded in a Euclidean space \mathbb{R}^8 . Thus, we can handle $\tilde{\sigma}$ as a vector with eight elements, $\tilde{\sigma} = [\hat{\mathbf{p}}^T, \hat{\mathbf{q}}^T]^T \in \mathbb{R}^8$. Moreover, the multiplication of the quaternion can be denoted in matrix expressions as

$$\hat{\mathbf{p}} \otimes \hat{\mathbf{q}} = [\hat{\mathbf{p}}]_{\otimes} \hat{\mathbf{q}} = [\hat{\mathbf{p}}]_{\otimes}^* \hat{\mathbf{q}} \quad (3)$$

where $[\hat{\mathbf{p}}]_{\otimes} = \begin{bmatrix} \mathbf{p} & p_0 I_3 + \mathbf{p}^x \\ p_0 & -\mathbf{p}^T \end{bmatrix}$ and $[\hat{\mathbf{q}}]_{\otimes}^* = \begin{bmatrix} \mathbf{q} & q_0 I_3 - \mathbf{q}^x \\ q_0 & -\mathbf{q}^T \end{bmatrix}$. Here, \mathbf{p}^x and \mathbf{q}^x represent the matrices related to the cross product operation, i.e., $\mathbf{p}^x = \begin{bmatrix} 0 & -p_3 & p_2 \\ p_3 & 0 & -p_1 \\ -p_2 & p_1 & 0 \end{bmatrix}$ and $\mathbf{q}^x = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$.

With the definition in (3), the dual quaternion multiplication can be expressed as

$$\tilde{\sigma}_1 \otimes \tilde{\sigma}_2 = [\tilde{\sigma}_1]_{\otimes} \tilde{\sigma}_2 = [\tilde{\sigma}_1]_{\otimes}^* \tilde{\sigma}_2 \quad (4)$$

where $[\tilde{\sigma}_1]_{\otimes} = \begin{bmatrix} [\hat{\mathbf{p}}_1]_{\otimes} & 0_{4 \times 4} \\ [\hat{\mathbf{q}}_1]_{\otimes} & [\hat{\mathbf{p}}_1]_{\otimes} \end{bmatrix}$ and $[\tilde{\sigma}_2]_{\otimes}^* = \begin{bmatrix} [\hat{\mathbf{p}}_2]_{\otimes}^* & 0_{4 \times 4} \\ [\hat{\mathbf{q}}_2]_{\otimes}^* & [\hat{\mathbf{p}}_2]_{\otimes}^* \end{bmatrix}$. Similarly, the cross operation of two quaternions can be expressed as matrix-vector multiplication

$$\hat{\mathbf{p}} \oslash \hat{\mathbf{q}} = [\hat{\mathbf{p}}]_{\oslash} \hat{\mathbf{q}} = [\hat{\mathbf{p}}]_{\oslash}^* \hat{\mathbf{q}} \quad (5)$$

where $[\hat{\mathbf{p}}]_{\oslash} = \begin{bmatrix} \mathbf{p} & p_0 I_3 + \mathbf{p}^x \\ 0 & 0_{1 \times 3} \end{bmatrix}$ and $[\hat{\mathbf{q}}]_{\oslash}^* = \begin{bmatrix} \mathbf{q} & q_0 I_3 - \mathbf{q}^x \\ 0 & 0_{1 \times 3} \end{bmatrix}$. With the definition in (5), the dual quaternion cross product operation can also be expressed as

$$\tilde{\sigma}_1 \oslash \tilde{\sigma}_2 = [\tilde{\sigma}_1]_{\oslash} \tilde{\sigma}_2 = [\tilde{\sigma}_1]_{\oslash}^* \tilde{\sigma}_2 \quad (6)$$

where $[\tilde{\sigma}_1]_{\oslash} = \begin{bmatrix} [\hat{\mathbf{p}}_1]_{\oslash} & 0_{4 \times 4} \\ [\hat{\mathbf{q}}_1]_{\oslash} & [\hat{\mathbf{p}}_1]_{\oslash} \end{bmatrix}$ and $[\tilde{\sigma}_2]_{\oslash}^* = \begin{bmatrix} [\hat{\mathbf{p}}_2]_{\oslash}^* & 0_{4 \times 4} \\ [\hat{\mathbf{q}}_2]_{\oslash}^* & [\hat{\mathbf{p}}_2]_{\oslash}^* \end{bmatrix}$.

B. Definition of Coordinate Frames

To describe the motion of spacecraft in the entry phase, the following five coordinate frames are defined as below and the relationship between them is shown in Fig. 1.

1) *Inertial Coordinate Frame $\mathcal{I} - O_I X_I Y_I Z_I$* : The origin O_I locates at the center of Mars, $O_I Z_I$ points to the North Pole direction. $O_I X_I$ and $O_I Y_I$ are in the Equatorial plane and determined by the right-hand rule.

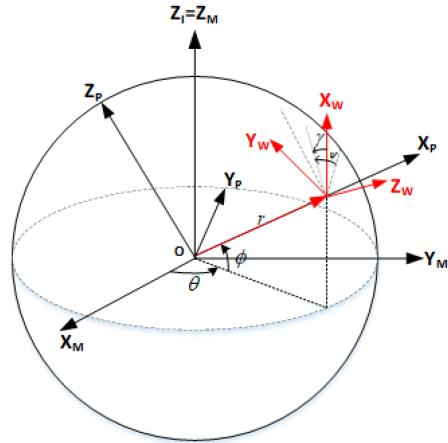


Fig. 1. Definition of coordinate frames.

2) *Mars-Fixed Coordinate Frame $\mathcal{M} - O_M X_M Y_M Z_M$* : The origin O_M locates at the center of Mars, $O_M Z_M$ points to the North Pole direction, $O_M X_M$ and $O_M Y_M$ rotate with Mars.

3) *Vehicle-Pointing Coordinate Frame $\mathcal{P} - O_P X_P Y_P Z_P$* : The origin O_P locates at the center of Mars, the X_P -axis-positive is pointing along the vehicle's position vector \vec{r} ; the Y_P -axis locates in the equatorial plane, and is perpendicular to the X_P -axis and parallel to the East of vehicle's subsatellite point; and the Z_P -axis is perpendicular to the $X_P - Y_P$ plane with the positive direction determined by the right-hand rule.

4) *Body Coordinate Frame $\mathcal{B} - O_B X_B Y_B Z_B$* : The origin O_B locates at the vehicle's center of gravity, the X_B -axis-positive points the nose of the vehicle in the plane of symmetry; the Z_B -axis is perpendicular to the X_B -axis, in the plane of symmetry of the vehicle; and the Y_B -axis is perpendicular to the $X_B - Z_B$ plane with positive determined by the right-hand rule.

5) *Wind Coordinate Frame $\mathcal{W} - O_W X_W Y_W Z_W$* : The origin O_W locates at the vehicle's center of gravity, the X_W -axis-positive is in the direction of the velocity vector of the vehicle relative to the air; the Y_W -axis positive is in the same direction as the projection of the lift force on the plane determined by X_P and velocity of the vehicle, and the Z_W -axis is perpendicular to the $X_W - Y_W$ plane with the positive direction determined by the right-hand rule.

III. PROBLEM FORMULATION

A. Dual-Quaternion-Based Kinematics and Dynamics of Atmospheric Entry Vehicle

The 6-DoF equations of motion based on dual quaternions over a spherical, rotating Mars are introduced in the following:

1) *6-DoF Kinematics*: We use the following dual quaternion components to denote the rotation and translation between inertial frame \mathcal{I} and body frame \mathcal{B} :

$$\tilde{\mathbf{q}}_{IB} = \hat{\mathbf{q}}_{rIB} + \epsilon \hat{\mathbf{q}}_{dIB} = \hat{\mathbf{q}}_{rIB} + \epsilon \left(\frac{1}{2} \hat{\mathbf{r}}_I \otimes \hat{\mathbf{q}}_{rIB} \right) \quad (7)$$

where $\hat{\mathbf{q}}_{dIB} \in \mathbb{H}$ denotes a translation $\hat{\mathbf{r}}_I \in \mathbb{H}$, expressed in the inertial frame \mathcal{I} , followed by a rotation $\hat{\mathbf{q}}_{rIB} \in \mathbb{H}$ with $\|\hat{\mathbf{q}}_{rIB}\| = 1$. Accordingly, the time derivatives of the dual quaternion elements can be written as

$$\dot{\hat{\mathbf{q}}}_{IB} = \frac{1}{2} \tilde{\mathbf{q}}_{IB} \otimes \tilde{\mathbf{w}}_{IB} \quad (8)$$

where $\tilde{\mathbf{w}}_{IB} = \hat{\omega}_{IB} + \epsilon \hat{\mathbf{v}}_I \in \mathbb{Q}$, $\hat{\omega}_{IB}$ and $\hat{\mathbf{v}}_I$ are pure quaternions representing the angular velocity and the linear velocity of the vehicle, respectively. For the simplicity of notation, $\tilde{\mathbf{q}}_{IB}$, $\hat{\mathbf{q}}_{rIB}$, $\hat{\mathbf{q}}_{dIB}$, $\tilde{\mathbf{w}}_{IB}$, and $\hat{\omega}_{IB}$ are written as $\tilde{\mathbf{q}}$, $\hat{\mathbf{q}}_r$, $\hat{\mathbf{q}}_d$, $\tilde{\mathbf{w}}$, and $\hat{\omega}$, respectively.

2) 6-DoF Dynamics: The vehicle dynamics during the entry phase is expressed as

$$\frac{d}{dt}(m\mathbf{v}_B) = m\dot{\mathbf{v}}_B + \boldsymbol{\omega}_B \times m\mathbf{v}_B = \mathbf{G}_B + \mathbf{F}_B \quad (9a)$$

$$\frac{d}{dt}(\mathbf{J}\boldsymbol{\omega}_B) = \mathbf{J}\dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times m\boldsymbol{\omega}_B = \mathbf{l}_w \times \mathbf{F}_B + \mathbf{M}_B = \mathbf{M}_N \quad (9b)$$

where $\boldsymbol{\omega}_B \in \mathbb{R}^{3 \times 1}$ is the projection of angular velocity of the vehicle on the body frame, $\mathbf{v}_B \in \mathbb{R}^{3 \times 1}$ is the projection of velocity on the body frame, m denotes the mass of the vehicle, $\mathbf{l}_w \in \mathbb{R}^{3 \times 1}$ denotes the constant body-frame vector from the vehicle's center of mass to the aerodynamic center, $\mathbf{F}_B \in \mathbb{R}^{3 \times 1}$ represents the aerodynamic force expressed in the body frame, $\mathbf{G}_B \in \mathbb{R}^{3 \times 1}$ is the gravity expressed in the body frame, $\mathbf{J} \in \mathbb{R}^{3 \times 3}$ represents the general inertia matrix of the entry vehicle, $\mathbf{M}_B \in \mathbb{R}^{3 \times 1}$ is the vector of moments generated by actuators, and $\mathbf{M}_N = [M_x, M_y, M_z]^T \in \mathbb{R}^{3 \times 1}$ is the vector of net moments that are handled as the control vector in the entry trajectory optimization problem. Combining (9a) and (9b), we can rewrite the dynamics using dual quaternions [18] as

$$\mathbf{J}_d \dot{\tilde{\mathbf{w}}} + \tilde{\mathbf{w}} \oslash \mathbf{J}_d \tilde{\mathbf{w}} = \Phi \tilde{\mathbf{F}}_B + \tilde{\mathbf{G}}_B + \tilde{\mathbf{M}}_N \quad (10)$$

where

$$\mathbf{J}_d = \left[\begin{array}{c|c} \mathbf{0}_{4 \times 4} & m\mathbf{I}_4 \\ \hline 1 & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 1} & \mathbf{J} \\ \hline & \mathbf{0}_{4 \times 4} \end{array} \right]_{8 \times 8} \in \mathbb{R}^{8 \times 8}$$

$$\Phi = \left[\begin{array}{c|c} \mathbf{0}_{8 \times 5} & \mathbf{l}_w^x \\ \hline & \mathbf{0}_{5 \times 3} \end{array} \right]_{8 \times 8} \in \mathbb{R}^{8 \times 8}$$

$$\tilde{\mathbf{F}}_B = \hat{\mathbf{0}} + \epsilon [0, \mathbf{F}_B^T]^T \in \mathbb{Q}, \quad \tilde{\mathbf{G}}_B = [0, \mathbf{G}_B^T]^T + \epsilon (\hat{\mathbf{0}}) \in \mathbb{Q}$$

$$\tilde{\mathbf{M}}_N = \hat{\mathbf{0}} + \epsilon [0, \mathbf{M}_N^T]^T \in \mathbb{Q}.$$

Here, \mathbf{I}_n is an n -dimensional identity matrix. Based on the definition of coordinate frames, the gravity and aerodynamic forces can be easily described in the vehicle-pointing frame \mathcal{P} and the wind frame \mathcal{W} , respectively. Specifically, the dual quaternion of gravity force in \mathcal{P} is written as $\tilde{\mathbf{G}}_P = \hat{\mathbf{0}} + \epsilon [0, \mathbf{G}_P^T]^T \in \mathbb{Q}$, where $\mathbf{G}_P = [-mg, 0, 0]^T$ and $g \in \mathbb{R}$ is the gravitational acceleration. Then, we have the relationship between $\tilde{\mathbf{G}}_P$ and $\tilde{\mathbf{G}}_B$, expressed as $\tilde{\mathbf{G}}_B = \tilde{\mathbf{q}}_{pb}^* \otimes \tilde{\mathbf{G}}_P \otimes \tilde{\mathbf{q}}_{pb}$, where $\tilde{\mathbf{q}}_{pb} \in \mathbb{Q}$ denotes the rotation and translation from \mathcal{P} frame to \mathcal{B} frame. Similarly, the dual quaternion of the aerodynamic forces in the wind frame \mathcal{W} can be written as

$$\tilde{\mathbf{F}}_W = \hat{\mathbf{0}} + \epsilon [0, \mathbf{F}_W^T]^T \in \mathbb{Q} \quad (11a)$$

$$\mathbf{F}_W = [-D, Y, L]^T \quad (11b)$$

$$= \left[-\frac{1}{2} C_D \rho S V^2, \frac{1}{2} C_Y \rho S V^2, \frac{1}{2} C_L \rho S V^2 \right]^T \quad (11c)$$

where $\rho \in \mathbb{R}$ is the Mars atmosphere density, $L \in \mathbb{R}$, $Y \in \mathbb{R}$, and $D \in \mathbb{R}$ are the lift, side, and drag forces, $S \in \mathbb{R}$ is the reference area, $V \in \mathbb{R}$ is the speed of vehicle relative to Mars' atmosphere, and $C_L \in \mathbb{R}$, $C_D \in \mathbb{R}$, $C_Y \in \mathbb{R}$ denote the lift, side, and drag coefficients of the vehicle, respectively. Moreover, C_L and C_D are linear and quadratic functions $\sin \alpha$, where α is the angle of attack, expressed as

$$C_L = c_{l_0} + c_{l_1} \sin \alpha, \quad C_D = c_{d_0} + c_{d_1} \sin \alpha + c_{d_2} \sin^2 \alpha \quad (12)$$

where $c_{l_0}, c_{l_1}, c_{d_0}, c_{d_1}$, and c_{d_2} are constant coefficients. Similarly, C_Y is a linear function of $\sin \beta$, where β is the side slip angle.

$$C_Y = c_{y_0} + c_{y_1} \sin \beta \quad (13)$$

where c_{y_0} and c_{y_1} are constant coefficients. Then, transforming $\tilde{\mathbf{F}}_W$ from the wind frame \mathcal{W} to the body frame \mathcal{B} , we have

$$\tilde{\mathbf{F}}_B = \tilde{\mathbf{q}}_{bw} \otimes \tilde{\mathbf{F}}_W \otimes \tilde{\mathbf{q}}_{bw}^* \quad (14)$$

where $\tilde{\mathbf{q}}_{bw} \in \mathbb{H}$ denotes the dual quaternion from \mathcal{B} frame to \mathcal{W} frame, which is defined as

$$\tilde{\mathbf{q}}_{bw} = \hat{\mathbf{q}}_{bw} + \epsilon \left(\frac{1}{2} \hat{\mathbf{l}}_w \otimes \hat{\mathbf{q}}_{bw} \right) \quad (15)$$

where $\hat{\mathbf{l}}_w \in \mathbb{H}$ is a pure quaternion expressed as $\hat{\mathbf{l}}_w = [0, \mathbf{l}_w^T]^T$, and the position vector \mathbf{l}_w of the aerodynamic center is assumed to be constant in the body frame. In fact, all four components in quaternion $\hat{\mathbf{q}}_{bw}$ are determined by two rotational angles (α, β) , expressed as

$$\hat{\mathbf{q}}_{bw} = \begin{bmatrix} q_{bw_1} \\ q_{bw_2} \\ q_{bw_3} \\ q_{bw_4} \end{bmatrix} = \begin{bmatrix} \cos \frac{\alpha}{2} \cos \frac{\beta}{2} \\ \sin \frac{\alpha}{2} \cos \frac{\beta}{2} \\ \cos \frac{\alpha}{2} \sin \frac{\beta}{2} \\ -\cos \frac{\alpha}{2} \sin \frac{\beta}{2} \end{bmatrix}. \quad (16)$$

Then, $\tilde{\mathbf{F}}_W$ can be expressed with respect to the dual angular velocity $\tilde{\mathbf{w}}$, written as

$$\tilde{\mathbf{F}}_W = \frac{1}{2} S \rho V^2 \Psi = \frac{1}{2} S \rho \|A \tilde{\mathbf{w}}\|^2 \tilde{\Psi} \quad (17)$$

where

$$\tilde{\Psi} = \hat{\mathbf{0}} + \epsilon [0, \Psi^T]^T \in \mathbb{Q}, \quad \Psi = [-C_D, C_Y, C_L]^T, \quad \text{and} \quad A = \begin{bmatrix} \mathbf{0}_{4 \times 4} \mathbf{0}_{4 \times 4} \\ \hline \mathbf{0}_{4 \times 4} \mathbf{l}_4 \end{bmatrix}_{8 \times 8} \in \mathbb{R}^{8 \times 8}.$$

B. Operational and Mission Constraints

For safe operation, specific operational and mission constraints during the entry phase are considered, including the following:

1) Stagnation-point convective heating load constraint

$$\dot{Q} = k_Q \sqrt{\frac{\rho}{R_{\text{nose}}}} V^{3.15} \leq \dot{Q}_{\max} \quad (18)$$

where $R_{\text{nose}} \in \mathbb{R}$ is the nose radius of the vehicle, $k_Q \in \mathbb{R}$ is a constant depending on the composition of the Martian atmosphere, and $\dot{Q}_{\max} \in \mathbb{R}$ denotes the allowable peak heating rate. This inequality constrains the heating rate at a stagnation point on the surface of the vehicle with a curvature radius R_{nose} [6].

2) Normal load constraint

$$\frac{\|\hat{\mathbf{F}}_W\|}{mg_0} = \frac{\sqrt{L^2 + D^2 + Y^2}}{mg_0} \leq n_{\max} \quad (19)$$

where $n_{\max} \in \mathbb{R}$ is the load factor for the allowable normal load on the surface of the entry vehicle, and $g_0 \in \mathbb{R}$ is the gravitational acceleration on the surface of Mars.

- 3) In this article, instead of using the conventional bank angle control with one control variable or direct force control with two control variables, we consider the control on the net moments (with three control variables) to further improve the flexibility in searching for the optimal entry trajectories. Then, the angle of attack and side slip angle are unfixed during the entry phase. Therefore, nonreaction control systems are assumed to be implemented in the entry vehicle that employs the aerodynamic surfaces, referred to as flaps, as the actuation systems [42], [43]. Then, the moments generated by the actuators are flight condition dependent and physically bounded, which indicates that the net moments are also bounded. We assume constant upper and lower bounds on the net moments such that

$$\begin{aligned} M_x^L &\leq M_x \leq M_x^U, \quad M_y^L \leq M_y \leq M_y^U \\ M_z^L &\leq M_z \leq M_z^U. \end{aligned} \quad (20)$$

Physically, the bounds on the net moment constrain the angular accelerations to avoid the dramatic changes of the vehicle's angular velocities and the attitudes, which prevents loss of control of the entry vehicle.

- 4) To avoid flow impingement and large radiative heating effects on the payload at a high angle of attack and side slip angle, a box constraint is considered for α and β , respectively,

$$S_\alpha^L \leq \sin \alpha \leq S_\alpha^U, \quad S_\beta^L \leq \sin \beta \leq S_\beta^U. \quad (21)$$

C. Additional Constraints

According to (7), the dual quaternion $\tilde{\mathbf{q}}$ can be divided into two parts $\hat{\mathbf{q}}_r$ and $\hat{\mathbf{q}}_d$, where $\hat{\mathbf{q}}_r$ represents the rotation from \mathcal{I} frame to \mathcal{B} frame. Here, we introduce $\hat{\mathbf{q}}_{\text{im}} \in \mathbb{H}$ to represent the rotation from \mathcal{I} frame to \mathcal{M} frame, $\hat{\mathbf{q}}_{\text{mp}} \in \mathbb{H}$ to represent the rotation from \mathcal{M} frame to \mathcal{P} frame, and $\hat{\mathbf{q}}_{\text{pb}} \in \mathbb{H}$ to represent the rotation from \mathcal{P} frame to \mathcal{B} frame. Then, according to the Euler's rotation theorem, we have

$$\hat{\mathbf{q}}_{\text{im}} \otimes \hat{\mathbf{q}}_{\text{mp}} \otimes \hat{\mathbf{q}}_{\text{pb}} = \hat{\mathbf{q}}_r \quad (22)$$

where $\hat{\mathbf{q}}_{\text{im}} = [\cos(\Omega t/2), 0, 0, \sin(\Omega t/2)]^T$, Ω is the rotational speed of Mars, and t is the entry time. By introducing the dual quaternion term $\hat{\mathbf{q}}_{\text{im}}$, the Mars rotation effects are

included in the dynamics without involving trigonometric functions. Next, we try to find the constraints on $\hat{\mathbf{q}}_{\text{mp}}$ based on the expression of $\hat{\mathbf{r}}_M$, which can be expressed as $\hat{\mathbf{r}}_M = [0, \mathbf{r}_M^T]^T$. Here, $\mathbf{r}_M \in \mathbb{R}^3$ is the vector from the origin of the \mathcal{M} frame to the origin of the \mathcal{B} frame projected on the \mathcal{M} frame, and it can be expressed as $\mathbf{r}_M = [r_{M_1}, r_{M_2}, r_{M_3}]^T$. According to (7), $\hat{\mathbf{r}}_M$ can be written as

$$\hat{\mathbf{r}}_M = \hat{\mathbf{q}}_{\text{im}}^* \otimes \hat{\mathbf{r}}_I \otimes \hat{\mathbf{q}}_{\text{im}} \quad (23)$$

where $\hat{\mathbf{r}}_I = 2\hat{\mathbf{q}}_d \otimes \hat{\mathbf{q}}_r^*$. Thus, we can obtain $r = \|\hat{\mathbf{r}}_I\|_2 = \|2\hat{\mathbf{q}}_d\|$. Here, $r \in \mathbb{R}$ is the radial distance between the gravity center of the vehicle and the center of Mars.

In addition, when considering the Mars rotation, there exists a relationship between the linear velocity $\hat{\mathbf{v}}_I$ and the velocity relative to the atmosphere, i.e., $\hat{\mathbf{v}}_M$, in the \mathcal{M} frame as

$$\hat{\mathbf{v}}_M = \hat{\mathbf{v}}_I - \hat{\omega}_{\mathbf{m}} \otimes \hat{\mathbf{r}}_I \quad (24)$$

where $\hat{\omega}_{\mathbf{m}} = [0, 0, 0, \Omega]$ is the quaternion that denotes the angular velocity between \mathcal{I} frame and \mathcal{M} frame. By projecting the velocity $\hat{\mathbf{v}}_M$ on the \mathcal{W} frame, we have

$$\hat{\mathbf{v}}_M = \hat{\mathbf{q}}_{\text{iw}} \otimes \hat{\mathbf{v}}_W \otimes \hat{\mathbf{q}}_{\text{iw}}^* \quad (25)$$

where $\hat{\mathbf{v}}_W = [0, V, 0, 0]^T$ and $\hat{\mathbf{q}}_{\text{iw}}$ is determined by

$$\hat{\mathbf{q}}_{\text{iw}} = \hat{\mathbf{q}}_r \otimes \hat{\mathbf{q}}_{\text{bw}}. \quad (26)$$

Besides, according to the definition of the coordinate frames, the coupling relationships for velocity \mathbf{v}_B , angle of attack α , and side-slip angle β can be written as

$$v_{Bx} \sin \alpha = -v_{Bz} \cos \alpha \quad (27)$$

$$v_{Bxy} \sin \beta = v_{By} \cos \beta \quad (28)$$

where $\mathbf{v}_B = [v_{Bx}, v_{By}, v_{Bz}]$, $v_{Bxy}^2 = v_{Bx}^2 + v_{By}^2$, and v_{Bx}, v_{By}, v_{Bz} are also components of $\tilde{\mathbf{w}}$.

It is obvious that $\tilde{\mathbf{q}}, \tilde{\mathbf{w}}, \hat{\mathbf{F}}_B, \hat{\mathbf{G}}_B, \hat{\mathbf{q}}_{\text{pb}}, \hat{\mathbf{r}}_M$, and $\hat{\mathbf{q}}_{\text{bw}}$ can all be updated by constraints (7)–(17) and (22)–(28). Here, we also need to show that $\hat{\mathbf{q}}_{\text{mp}}$ can also be updated based on $\tilde{\mathbf{q}}, \tilde{\mathbf{w}}, \hat{\mathbf{F}}_B, \hat{\mathbf{G}}_B, \hat{\mathbf{q}}_{\text{pb}}, \hat{\mathbf{r}}_M$, and $\hat{\mathbf{q}}_{\text{bw}}$. Moreover, by introducing $\hat{\mathbf{q}}_{\text{mp}}$, we have

$$\hat{\mathbf{r}}_M = \hat{\mathbf{q}}_{\text{mp}} \otimes \hat{\mathbf{r}}_P \otimes \hat{\mathbf{q}}_{\text{mp}}^*. \quad (29)$$

From (29), it holds that

$$\hat{\mathbf{r}}_M = \hat{\mathbf{q}}_{\text{mp}} \otimes \hat{\mathbf{r}}_P \otimes \hat{\mathbf{q}}_{\text{mp}}^* = \begin{bmatrix} 0 \\ r \left(q_{\text{mp}_1}^2 + q_{\text{mp}_2}^2 - q_{\text{mp}_3}^2 - q_{\text{mp}_4}^2 \right) \\ 2r \left(q_{\text{mp}_1} q_{\text{mp}_4} + q_{\text{mp}_2} q_{\text{mp}_3} \right) \\ 2r \left(q_{\text{mp}_2} q_{\text{mp}_4} - q_{\text{mp}_1} q_{\text{mp}_3} \right) \end{bmatrix}.$$

Since $\hat{\mathbf{q}}_{\text{mp}}$ is determined by the longitude $\theta \in \mathbb{R}$ and latitude $\phi \in \mathbb{R}$. Then, $\hat{\mathbf{q}}_{\text{mp}}$ can be written as

$$\hat{\mathbf{q}}_{\text{mp}} = \left[\cos \frac{\theta}{2} \cos \frac{\phi}{2}, \sin \frac{\theta}{2} \cos \frac{\phi}{2}, \sin \frac{\theta}{2} \sin \frac{\phi}{2}, -\cos \frac{\theta}{2} \sin \frac{\phi}{2} \right]^T. \quad (30)$$

Substituting (30) into $\hat{\mathbf{r}}_M$, we can obtain the following relationships:

$$\begin{aligned} r_{M_1} &= r \cos \phi, \quad r_{M_2} = -r \cos \theta \sin \phi, \\ r_{M_3} &= -r \sin \theta \sin \phi. \end{aligned} \quad (31)$$

Then, ϕ and θ can be determined by

$$\phi = \arccos\left(\frac{r_{M_1}}{r}\right), \quad \theta = \arctan\left(-\frac{r_{M_3}}{r_{M_2}}\right) \quad (32)$$

with given θ and ϕ , $\hat{\mathbf{q}}_{\text{mp}}$ is determined accordingly.

D. Formulation of the Entry Trajectory Optimization Problem

In this subsection, we organize the aforementioned constraints during the entry phase and formulate the entry trajectory optimization problem. The objective is to minimize the terminal speed at a specified terminal altitude while satisfying all constraints stated previously. Mathematically, the entry trajectory optimization problem is formulated as

$$\begin{aligned} & \min_{M_x, M_y, M_z, t_f} \| \mathbf{A} \tilde{\mathbf{w}}(t_f) \| \\ & \text{subject to} \\ & \dot{\tilde{\mathbf{q}}} = \frac{1}{2} \tilde{\mathbf{q}} \otimes \tilde{\mathbf{w}}, \mathbf{J}_d \dot{\tilde{\mathbf{w}}} + \tilde{\mathbf{w}} \otimes \mathbf{J}_d \tilde{\mathbf{w}} = \Phi \tilde{\mathbf{F}}_B + \tilde{\mathbf{G}}_B + \tilde{\mathbf{M}}_B \\ & \tilde{\mathbf{G}}_B = \tilde{\mathbf{q}}_{\text{pb}}^* \otimes \tilde{\mathbf{G}}_P \otimes \tilde{\mathbf{q}}_{\text{pb}}, \quad \tilde{\mathbf{q}} = \tilde{\mathbf{q}}_{\text{im}} \otimes \tilde{\mathbf{q}}_{\text{mp}} \otimes \tilde{\mathbf{q}}_{\text{pb}} \\ & \tilde{\mathbf{F}}_B = \tilde{\mathbf{q}}_{\text{wb}}^* \otimes \tilde{\mathbf{F}}_W \otimes \tilde{\mathbf{q}}_{\text{wb}}, \quad \tilde{\mathbf{F}}_W = \frac{1}{2} S \rho \| \mathbf{A} \tilde{\mathbf{w}} \|^2 \Psi \\ & (18), (19), (20), (21), (12), (13), (27), (28) \\ & \tilde{\mathbf{q}}(t_0) = \mathbf{q}_0, \quad \tilde{\mathbf{w}}(t_0) = \tilde{\mathbf{w}}_0 \quad \| \tilde{\mathbf{q}}(t_f) \| = R_f \end{aligned} \quad (33)$$

where $\tilde{\mathbf{G}}_P = [\mathbf{0}_{1 \times 4}, 0, -mg, 0, 0]^T$, t_0 and t_f are, respectively, the starting and final time of the entry phase, and R_f is the terminal radial distance.

Due to the specific entry trajectory optimization problem formulated in (33), its optimal control profile has unique properties.

PROPOSITION III.1 In the 6-DoF dual-quaternion-based entry trajectory optimization problem formulated in (33), the optimal solution of each moment component, M_x , M_y , and M_z , has a bang–bang control profile.

PROOF The details of the proof can be found in Appendix A.

Although Proposition III.1 proves that the optimal solution has bang–bang control profiles, the existence of nonlinear constraints makes it difficult to derive an explicit optimal solution using Pontryagin’s maximum principle. Therefore, a hybrid ADMM is proposed to solve the entry trajectory optimization problem.

E. Equivalent Formulation Based on Flight-Path Coordinates

In order to demonstrate the correctness and advantages of the formulation based on dual quaternion, we reformulate the entry trajectory optimization problem in (33) using the traditional entry dynamical model based on the flight-path coordinates. The translational motion of an entry vehicle based on the flight-path coordinates is reformulated as follows:

$$\dot{r} = V \sin(\gamma), \quad \dot{\theta} = \frac{V \cos \gamma \cos \psi}{r \cos \phi}, \quad \dot{\phi} = \frac{V \cos \gamma \sin \psi}{r}$$

$$\begin{aligned} \dot{V} &= -D - \frac{V \sin \gamma}{r^2} \\ &\quad - r \Omega^2 \cos \phi (\cos \phi \sin \gamma + \sin \phi \sin \psi \cos \gamma) \\ \dot{\gamma} &= \frac{1}{V} \left(L \cos \sigma + Y \sin \sigma + \left(V^2 - \frac{1}{r} \right) \frac{\cos \gamma}{r} \right) \\ &\quad + 2\Omega \cos \phi \cos \psi + \frac{\Omega^2 r \cos \phi (\cos \phi \cos \gamma - \sin \phi \sin \psi \sin \gamma)}{V} \\ \dot{\psi} &= \frac{L \sin \sigma - Y \cos \sigma}{V \cos \gamma} - \frac{V \cos \gamma \cos \psi \tan \phi}{r} \\ &\quad - 2\Omega (\sin \psi \cos \phi \tan \gamma + \sin \phi) - \frac{\Omega^2 r \sin \phi \cos \phi \cos \psi}{V \cos \gamma}. \end{aligned} \quad (34)$$

In the traditional entry dynamical model based on the flight-path coordinates, only the translational motion is considered. To ensure the equivalence of the two model, the rotational motion formulations for the two problems are both represented by quaternions in (9b) and (11a)–(16). Accordingly, the time derivatives of the quaternion $\hat{\mathbf{q}}_r$ can be written as

$$\dot{\hat{\mathbf{q}}}_r = \frac{1}{2} \hat{\mathbf{q}}_r \otimes \hat{\boldsymbol{\omega}}_B. \quad (35)$$

Here, we introduce $\hat{\mathbf{q}}_{\text{pw}} \in \mathbb{H}$ to represent the rotation from \mathcal{P} frame to \mathcal{W} frame. Based on the updated angles γ , ψ , and σ in (34), the quaternion $\hat{\mathbf{q}}_{\text{pw}}$ can be represented by

$$\hat{\mathbf{q}}_{\text{pw}} = \begin{bmatrix} \cos \psi \\ 0 \\ 0 \\ \sin \psi \end{bmatrix} \otimes \begin{bmatrix} \cos \gamma \\ 0 \\ \sin \gamma \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos \sigma \\ \sin \sigma \\ 0 \\ 0 \end{bmatrix}. \quad (36)$$

According to (16), (22), (26), (30), (36), angle of attack α and side-slip angle β can be determined. Then, the 6-DoF entry trajectory optimization problem based on flight-path coordinates can be formulated as

$$\begin{aligned} & \min_{M_x, M_y, M_z, t_f} V(t_f) \\ & \text{subject to} \\ & (34), (35), (36), (9b), (11a) – (16), (22), (26), (30) \\ & (18), (19), (20), (21), (12), (13), (27), (28), \\ & r(t_0) = r_0, V(t_0) = V_0, \theta(t_0) = \theta_0, \phi(t_0) = \phi_0, \gamma(t_0) = \gamma_0 \\ & \psi(t_0) = \psi_0, \quad \hat{\mathbf{q}}_r(t_0) = \hat{\mathbf{q}}_{r0}, \quad \hat{\boldsymbol{\omega}}_B(t_0) = \hat{\boldsymbol{\omega}}_{B0}, \quad r(t_f) = r_f. \end{aligned} \quad (37)$$

The 6-DoF entry trajectory optimization problem in (37) will be solved using an NLP solver for the comparison purpose, which will be presented in Section VI.

IV. PROBLEM CONVERSION INTO A QCQP

A. Discretization and Conversion Into a QCQP

In the dual-quaternion-based entry trajectory optimization problem formulated in (33), most constraints are formulated as quadratic equalities or inequalities, except for those involving nonpolynomial terms, e.g., the exponential terms in the heating load constraint and the atmosphere density function. These nonpolynomial functions will be approximated by high-order polynomials with negligible fitting errors. Specifically, the normalized nonlinear atmosphere

density is approximated by a six-order polynomial function with the maximum error less than 1e-4, expressed as

$$\rho = \sum_{i=0}^{n=6} p_i h^i \quad (38)$$

where p_i , $i = 0, \dots, 6$, are the fitting coefficients, and $h = (r - R_m) = \|\tilde{\mathbf{q}}\| - R_m$ is the altitude.

For the maximum heating rate constraint in (18), it can be rewritten as

$$V \leq \sqrt[3.15]{\frac{Q_{\max}}{k_Q \sqrt{\rho}/R_{\text{nose}}}} := V_{\max}. \quad (39)$$

Similarly, V_{\max} can be approximated by a fourth-order polynomial function

$$V_{\max} = \sum_{i=0}^{n=4} u_i h^i \quad (40)$$

where u_i , $i = 0, \dots, 4$, are the fitting coefficients.

Through the approximations in (38) and (40), the entry trajectory optimization problem in (33) can be reformulated as a polynomial optimal control problem. It is known that a polynomial optimal control problem can be converted into a polynomial programming problem via discretization techniques [44]. Then, by introducing intermediate variables and quadratic constraints, it can be equivalently reformulated as a homogeneous QCQP, and more details about the reformulation of QCQP can be found in [38] and [44]. The general expression of QCQP is written as

$$\begin{aligned} & \min_{\mathbf{z} \in \mathbb{R}^n} \mathbf{z}^T \mathbf{A}_0 \mathbf{z} \\ & \text{subject to } \mathbf{z}^T \mathbf{A}_i \mathbf{z} = a_i, \quad i \in \mathcal{E} \\ & \quad \mathbf{z}^T \mathbf{B}_j \mathbf{z} \leq b_j, \quad j \in \mathcal{I} \end{aligned} \quad (41)$$

where $\mathbf{z} \in \mathbb{R}^n$ is the unknown vector to be determined, the coefficient matrices $\mathbf{A}_0 \in \mathbb{R}^{n \times n}$, $\mathbf{A}_i \in \mathbb{R}^{n \times n}$, $i \in \mathcal{E}$, and $\mathbf{B}_j \in \mathbb{R}^{n \times n}$, $j \in \mathcal{I}$, are real systematic and not necessarily to be positive semidefinite. \mathcal{E} and \mathcal{I} denote the indices sets of equality and inequality constraints, respectively. Due to the indefiniteness of \mathbf{A}_i or \mathbf{B}_j , (41) is generally a nonconvex problem and NP-hard to solve. In the following, a hybrid ADMM is proposed to solve large-scale nonconvex QCQPs.

V. HYBRID ADMM FOR NONCONVEX QCQPS

A. Framework of Hybrid ADMM

To solve the nonconvex QCQPs, (41) is first equivalently transformed into a consensus-constrained optimization problem, which is expressed as

$$\begin{aligned} & \min_{\mathbf{z}, \mathbf{x} \in \mathbb{R}^n} \mathbf{x}^T \mathbf{A}_0 \mathbf{z} \\ & \text{subject to } \mathbf{x}^T \mathbf{A}_i \mathbf{z} = a_i, \quad i \in \mathcal{E} \\ & \quad \mathbf{z}^T \mathbf{B}_j \mathbf{x} \leq b_j, \quad j \in \mathcal{I} \\ & \quad \mathbf{z} = \mathbf{x}. \end{aligned} \quad (42)$$

Obviously, the consensus constraint $\mathbf{z} = \mathbf{x}$ ensures the equivalence between problems (41) and (42). Let $\boldsymbol{\mu} \in \mathbb{R}^{|\mathcal{E}|}$, $\lambda \in \mathbb{R}^{|\mathcal{I}|}$, and $\boldsymbol{\nu} \in \mathbb{R}^n$ be the Lagrange multipliers associated with the equality constraints, inequality constraints, and the

consensus constraint in (42), respectively. Here, $|\mathcal{E}|$ and $|\mathcal{I}|$ denote the cardinalities of \mathcal{E} and \mathcal{I} , respectively. For notational convenience, we denote $\Lambda = [\boldsymbol{\nu}^T, \boldsymbol{\mu}^T, \lambda^T]^T$. Next, a logical function associated with the inequality constraint, $\mathbf{z}^T \mathbf{B}_j \mathbf{x} \leq b_j$, $j \in \mathcal{I}$, is defined as follows:

$$\Gamma_j(\mathbf{x}, \mathbf{z}, \lambda_j) = \begin{cases} 0, & \lambda_j + \zeta (\mathbf{x}^T \mathbf{B}_j \mathbf{z} - b_j) \leq 0 \\ 1, & \lambda_j + \zeta (\mathbf{x}^T \mathbf{B}_j \mathbf{z} - b_j) > 0 \end{cases} \quad (43)$$

where ζ denotes a positive constant. By employing (43) to handle the inequalities in (42), we have the augmented Lagrangian for (42), written as

$$\begin{aligned} \mathcal{L}_{\mathbf{p}}(\mathbf{x}, \mathbf{z}, \Lambda) = & \mathbf{x}^T \mathbf{A}_0 \mathbf{z} + \boldsymbol{\nu}^T (\mathbf{x} - \mathbf{z}) + \frac{\zeta_1}{2} \|\mathbf{x} - \mathbf{z}\|^2 \\ & + \sum_{i \in \mathcal{E}} \left(\mu_i (\mathbf{x}^T \mathbf{A}_i \mathbf{z} - a_i) + \frac{\zeta_2}{2} \|\mathbf{x}^T \mathbf{A}_i \mathbf{z} - a_i\|^2 \right) \\ & + \sum_{j \in \mathcal{I}} \Gamma_j \left(\lambda_j (\mathbf{x}^T \mathbf{B}_j \mathbf{z} - b_j) + \frac{\zeta_3}{2} \|\mathbf{x}^T \mathbf{B}_j \mathbf{z} - b_j\|^2 \right) \end{aligned} \quad (44)$$

where $\mathbf{p} = [\zeta_1, \zeta_2, \zeta_3]$ denote the collection of the penalty coefficients associated with all the augmented terms.

To satisfy the consensus constraint $\mathbf{x} = \mathbf{z}$, two penalty terms are introduced in the augmented Lagrangian in (44). The customized ADMM in [37], [38], and [39] requires many iterations to drive \mathbf{x} to \mathbf{z} with high accuracy, especially when the initial guess is not close to the optimal solution and the penalty coefficients associated with $(\mathbf{x} - \mathbf{z})$ and $\|\mathbf{x} - \mathbf{z}\|^2$ are not significantly large. However, assigning large penalty coefficients for the associated $(\mathbf{x} - \mathbf{z})$ and $\|\mathbf{x} - \mathbf{z}\|^2$ terms may make the augmented Lagrangian ill-balanced. To obtain fast convergence without sacrificing the optimality value, a two-stage updating law is proposed for the primary variables \mathbf{x} and \mathbf{z} . The first stage follows the original ADMM framework [45] that updates \mathbf{x} and \mathbf{z} alternatively. The second stage forces \mathbf{x} and \mathbf{z} to equal to their average after the primary updates. Then, a hybrid ADMM is expressed as

$$\mathbf{x}^{k+1} = \arg \min_{\mathbf{x}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}, \mathbf{z}^k, \Lambda^k) \quad (45a)$$

$$\mathbf{z}^{k+1} = \arg \min_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}, \Lambda^k) \quad (45b)$$

If $k \geq k_{\max}$ or \mathbf{x}^k satisfies a bounded error condition:

$$\mathbf{x}^{k+1} = \mathbf{z}^{k+1} = \frac{\mathbf{x}^{k+1} + \mathbf{z}^{k+1}}{2} \quad (45c)$$

$$\boldsymbol{\nu}^{k+1} = \boldsymbol{\nu}^k + \zeta_1^k (\mathbf{x}^{k+1} - \mathbf{z}^{k+1}) \quad (45d)$$

$$\mu_i^{k+1} = \mu_i^k + \zeta_2^k ((\mathbf{x}^{k+1})^T \mathbf{A}_i \mathbf{z}^{k+1} - a_i) \quad \forall i \in \mathcal{E} \quad (45e)$$

$$\lambda_j^{k+1} = \max \left\{ 0, \lambda_j^k + \zeta_3^k ((\mathbf{x}^{k+1})^T \mathbf{B}_j \mathbf{z}^{k+1} - b_j) \right\} \quad \forall j \in \mathcal{I} \quad (45f)$$

where the bounded error condition refers to that the relative residuals of equality, inequality, and consensus constraints

are smaller than a constant δ as

$$\|\xi\|_1 = \left\| \begin{array}{c} \sum_{i \in \mathcal{E}} \|(\mathbf{x}^{k+1})^T \mathbf{A}_i \mathbf{z}^{k+1} - a_i\| \\ \sum_{j \in \mathcal{I}} \|(\mathbf{x}^{k+1})^T \mathbf{B}_j \mathbf{z}^{k+1} - b_j\| \\ \frac{\|\mathbf{x} - \mathbf{z}\|}{\|\mathbf{x}\|} \end{array} \right\|_1 \leq \delta. \quad (46)$$

In addition, the penalty coefficients ξ_1^k , ξ_2^k , and ξ_3^k are assigned as nondecreasing positive sequences based on the following updating rules:

$$\xi_1^{k+1} = \begin{cases} \beta \xi_1^k, & \|\mathbf{x}^{k+1} - \mathbf{z}^{k+1}\| \geq \tau \|\mathbf{x}^k - \mathbf{z}^k\| \\ \xi_1^k, & \text{otherwise} \end{cases} \quad (47a)$$

$$\xi_2^{k+1} = \begin{cases} \beta \xi_2^k, & \|(\mathbf{x}^{k+1})^T \mathbf{A}_{\mathcal{E}} \mathbf{z}^{k+1} - \mathbf{a}\| \\ \geq \tau \|(\mathbf{x}^k)^T \mathbf{A}_{\mathcal{E}} \mathbf{z}^k - \mathbf{a}\| \\ \xi_2^k, & \text{otherwise} \end{cases} \quad (47b)$$

$$\xi_3^{k+1} = \begin{cases} \beta \xi_3^k, & \|[(\mathbf{x}^{k+1})^T \mathbf{B}_{\mathcal{I}} \mathbf{z}^{k+1} - \mathbf{b}]^+\| \\ \geq \tau \|[(\mathbf{x}^k)^T \mathbf{B}_{\mathcal{I}} \mathbf{z}^k - \mathbf{b}]^+\| \\ \xi_3^k, & \text{otherwise} \end{cases} \quad (47c)$$

where $\beta \geq 1$ and τ are positive constants, and $\mathbf{A}_{\mathcal{E}}$ is defined as

$$\mathbf{x}^T \mathbf{A}_{\mathcal{E}} \mathbf{z} - \mathbf{c} = \begin{bmatrix} \mathbf{x}^T \mathbf{A}_1 \mathbf{z} - a_1 \\ \mathbf{x}^T \mathbf{A}_2 \mathbf{z} - a_2 \\ \vdots \\ \mathbf{x}^T \mathbf{A}_{|\mathcal{E}|} \mathbf{z} - a_{|\mathcal{E}|} \end{bmatrix} \in \mathbb{R}^{|\mathcal{E}| \times 1}$$

$$\mathbf{A}_{\mathcal{E}} \mathbf{x} = [\mathbf{A}_1 \mathbf{x}, \mathbf{A}_2 \mathbf{x}, \dots, \mathbf{A}_{|\mathcal{E}|} \mathbf{x}] \in \mathbb{R}^{|\mathcal{E}| \times n}. \quad (48)$$

Similar definition is made for $\mathbf{B}_{\mathcal{I}}$ such that $\mathbf{B}_{\mathcal{I}} \mathbf{x} = [\mathbf{B}_1 \mathbf{x}, \mathbf{B}_2 \mathbf{x}, \dots, \mathbf{B}_{|\mathcal{I}|} \mathbf{x}] \in \mathbb{R}^{|\mathcal{I}| \times n}$ and $[(\mathbf{x}^{k+1})^T \mathbf{B}_{\mathcal{I}} \mathbf{z}^{k+1} - \mathbf{b}]^+ := \max\{(\mathbf{x}^{k+1})^T \mathbf{B}_{\mathcal{I}} \mathbf{z}^{k+1} - \mathbf{b}, -\lambda_j^{k+1}/\xi_3^k\}$.

For the \mathbf{x} -update in (45a), (44) turns out to be a ξ_1^k -strongly convex function when $\mathbf{z} = \mathbf{z}^k$ and $\Lambda = \Lambda^k$ are given. Thus, by applying the first-order optimality conditions of (44), we can derive the optimum of subproblem (45a) as

$$\begin{aligned} \frac{\partial \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}, \mathbf{z}^k, \Lambda^k)}{\partial \mathbf{x}} &= \mathbf{A}_0 \mathbf{z}^k + \mathbf{v}^k + \xi_1^k (\mathbf{x} - \mathbf{z}^k) \\ &\quad + \sum_{i \in \mathcal{E}} (\mu_i^k \mathbf{A}_i \mathbf{z}^k + \xi_2^k (\mathbf{x}^T \mathbf{A}_i \mathbf{z}^k - a_i) \mathbf{A}_i \mathbf{z}^k) \\ &\quad + \sum_{j \in \mathcal{I}} \Gamma_{xj}^k (\lambda_j^k \mathbf{B}_j \mathbf{z}^k + \xi_3^k (\mathbf{x}^T \mathbf{B}_j \mathbf{z}^k - b_j) \mathbf{B}_j \mathbf{z}^k) \\ &= \mathbf{0} \end{aligned} \quad (49)$$

where $\Gamma_{xj}^k = \Gamma_j(\mathbf{x}^k, \mathbf{z}^k, \lambda_j^k)$, $j \in \mathcal{I}$, denotes the logical function associated with the inequality constraint j at the k th step. Rearranging (49), we can find the analytical solution of the \mathbf{x} -update, expressed as

$$\mathbf{x}^{k+1} = (\mathbf{Q}_x^k)^{-1} \mathbf{t}_x^k \quad (50)$$

where \mathbf{Q}_x^k and \mathbf{t}_x^k are defined as

$$\mathbf{Q}_x^k = \xi_1^k \mathbf{I} + \sum_{i \in \mathcal{E}} \xi_2^k (\mathbf{A}_i \mathbf{z}^k) (\mathbf{A}_i \mathbf{z}^k)^T$$

$$+ \sum_{j \in \mathcal{I}} \Gamma_{xj}^k \xi_3^k (\mathbf{B}_j \mathbf{z}^k) (\mathbf{B}_j \mathbf{z}^k)^T \quad (51a)$$

$$\begin{aligned} \mathbf{t}_x^k &= -\mathbf{A}_0 \mathbf{z}^k - \mathbf{v}^k + \xi_1^k \mathbf{z}^k - \sum_{i \in \mathcal{E}} ((\mu_i^k - \xi_2^k a_i) \mathbf{A}_i \mathbf{z}^k) \\ &\quad - \sum_{j \in \mathcal{I}} (\Gamma_{xj}^k (\lambda_j^k - \xi_3^k b_j) \mathbf{B}_j \mathbf{z}^k) \end{aligned} \quad (51b)$$

where \mathbf{Q}_x^k is always a positive definite matrix by setting all positive elements in \mathbf{p}^k . Similarly, with a fixed \mathbf{x}^{k+1} and Λ^k , the analytical solution of the subproblem (45b) for the \mathbf{z} -update can be written as

$$\mathbf{z}^{k+1} = (\mathbf{Q}_z^k)^{-1} \mathbf{t}_z^k \quad (52)$$

where

$$\begin{aligned} \mathbf{Q}_z^k &= \xi_1^k \mathbf{I} + \sum_{i \in \mathcal{E}} \xi_2^k (\mathbf{A}_i \mathbf{x}^{k+1}) (\mathbf{A}_i \mathbf{x}^{k+1})^T \\ &\quad + \sum_{j \in \mathcal{I}} \Gamma_{zj}^k \xi_3^k (\mathbf{B}_j \mathbf{x}^{k+1}) (\mathbf{B}_j \mathbf{x}^{k+1})^T \end{aligned} \quad (53a)$$

$$\begin{aligned} \mathbf{t}_z^k &= -\mathbf{A}_0 \mathbf{x}^{k+1} + \mathbf{v}^k + \xi_1^k \mathbf{x}^{k+1} \\ &\quad - \sum_{i \in \mathcal{E}} ((\mu_i^k - \xi_2^k a_i) \mathbf{A}_i \mathbf{x}^{k+1}) \\ &\quad - \sum_{j \in \mathcal{I}} (\Gamma_{zj}^k (\lambda_j^k - \xi_3^k b_j) \mathbf{B}_j \mathbf{x}^{k+1}) \end{aligned} \quad (53b)$$

$$\Gamma_{zj}^k = \Gamma_j(\mathbf{x}^{k+1}, \mathbf{z}^k, \lambda_j^k). \quad (53c)$$

Since it is straightforward to update Λ^k in (45), with the closed-form solutions in (50) and (52) derived for the subproblems (45a) and (45b), the hybrid ADMM for a nonconvex QCQP in (41) is summarized in Algorithm 1.

B. Convergence Analysis of the Hybrid ADMM Algorithm

This section provides the convergence analysis of the hybrid ADMM. We first describe the “bounded relative error” of the solutions from (45a) and (45b), and then, show that this condition can be satisfied before applying the consensus updating laws. Second, we prove that after employing the consensus updates in the second stage, the hybrid ADMM will converge to a stationary point with a linear convergence rate.

DEFINITION V.1 (Bounded error condition). Given \mathbf{x}^k , \mathbf{z}^k , and Λ^k , we say that the subproblems in (45a) and (45b) are solved with a bounded relative error if there is a constant c such that

$$\mathbf{0} = \frac{\partial \mathcal{L}}{\partial \mathbf{z}}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^{k+1}) + \delta^{k+1} \quad (54a)$$

$$\begin{aligned} \|\delta^{k+1}\| &\leq c \left(\|\mathbf{v}^{k+1} - \mathbf{v}^k\| + \|\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k\| \right. \\ &\quad \left. + \|\Gamma_{\Lambda}^k \circ (\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k)\| \right) \end{aligned} \quad (54b)$$

where “ \circ ” denotes the element-wise product.

Remarkably, we do not need to check this condition at each iteration. Next, we show that for a fixed number of

Algorithm 1: Framework of Hybrid consensus ADMM for QCQPs.

Input: $\mathbf{A}_0, \mathbf{A}_i, \mathbf{a}_i, i \in \mathcal{E}, \mathbf{B}_j, \mathbf{b}_j, j \in \mathcal{I}$, and algorithmic parameters $\beta, \tau, \epsilon, \delta$

Output: Vectors \mathbf{x} and \mathbf{z}

Initialization: $\mathbf{x}^0 = \mathbf{z}^0, \Lambda^0$ and penalty coefficients \mathbf{p}^0

- 1: /* The first stage */
- 2: **for** $k = 0, 1, 2, \dots, k_{\max_1}$ **do**
- 3: Compute $\Gamma_{x_j}^k$ according to $\mathbf{x}^k, \mathbf{z}^k, \Lambda^k$;
- 4: Update \mathbf{x}^{k+1} using (50) with \mathbf{z}^k, Λ^k ;
- 5: Compute $\Gamma_{z_j}^k$ according to $\mathbf{x}^{k+1}, \mathbf{z}^k, \Lambda^k$;
- 6: Update \mathbf{z}^{k+1} using (52) with $\mathbf{x}^{k+1}, \Lambda^k$;
- 7: Update Λ using (45d), (45e), and (45f);
- 8: Compute ξ = $\begin{bmatrix} \frac{\sum_{i \in \mathcal{E}} \|(\mathbf{x}^{k+1})^T \mathbf{A}_i \mathbf{z}^{k+1} - a_i\|}{\|\mathbf{a}\|}, \frac{\sum_{j \in \mathcal{I}} \|(\mathbf{x}^{k+1})^T \mathbf{B}_j \mathbf{z}^{k+1} - b_j\|}{\|\mathbf{b}\|}, \\ \frac{\|\mathbf{x} - \mathbf{z}\|}{\|\mathbf{x}\|} \end{bmatrix}$
- 9: **if** $\|\xi\|_1 \leq \delta$ **then**
- 10: the bounded error is satisfied, break;
- 11: **end if**
- 12: Update penalty coefficients using (47);
- 13: **end for**
- 14: /* The second stage */
- 15: **Reinitialize** $\mathbf{x}^0 = \mathbf{z}^0 = \frac{\mathbf{x}^k + \mathbf{z}^k}{2}, \Lambda^0$, and \mathbf{p}^0 .
- 16: **for** $k = 0, 1, 2, \dots$ **do**
- 17: Compute $\Gamma_{x_j}^k$ according to $\mathbf{x}^k, \mathbf{z}^k, \Lambda^k$;
- 18: Update \mathbf{x}^{k+1} using (50) with \mathbf{z}^k, Λ^k ;
- 19: Compute $\Gamma_{z_j}^k$ according to $\mathbf{x}^{k+1}, \mathbf{z}^k, \Lambda^k$;
- 20: Update \mathbf{z}^{k+1} using (52) with $\mathbf{x}^{k+1}, \Lambda^k$;
- 21: Apply the consensus: $\mathbf{x}^{k+1} = \mathbf{z}^{k+1} = \frac{\mathbf{x}^{k+1} + \mathbf{z}^{k+1}}{2}$
- 22: Update Λ using (45d), (45e), and (45f);
- 23: Calculate the error vector ξ = $\begin{bmatrix} \frac{\sum_{i \in \mathcal{E}} \|(\mathbf{x}^{k+1})^T \mathbf{A}_i \mathbf{z}^{k+1} - a_i\|}{\|\mathbf{a}\|}, \frac{\sum_{j \in \mathcal{I}} \|(\mathbf{x}^{k+1})^T \mathbf{B}_j \mathbf{z}^{k+1} - b_j\|}{\|\mathbf{b}\|} \end{bmatrix}$
- 24: **if** $\|\xi\|_1 \leq \epsilon$ **then**
- 25: break;
- 26: **end if**
- 27: Update penalty coefficients using (47);
- 28: $k = k + 1$
- 29: **end for**

iterations, there is a constant c such that the condition is satisfied. We start with the following two assumptions.

Assumption V.2 The objective function and constraints of problem (41) are all twice differentiable. Moreover, the objective function is lower bounded and there exists at least one feasible solution for problem (41).

Assumption V.3 In problem (41), $\sum_{i \in \mathcal{E}} \mathbf{A}_i$ has a full column rank, that is, for any nonzero vector $\mathbf{x} \in \mathbb{R}^n$, $\text{rank}(\sum_{i \in \mathcal{E}} \mathbf{A}_i \mathbf{x}) \geq |\mathcal{E}|$. Moreover, $\sum_{j \in \mathcal{I}} \mathbf{B}_j$ also has a full column rank, that is, $\text{rank}(\sum_{j \in \mathcal{I}} \mathbf{B}_j \mathbf{x}) \geq |\mathcal{I}|$.

THEOREM V.4 Let Assumptions V.2 and V.3 hold, and $\{\mathbf{x}^k, \mathbf{z}^k, \Lambda^k\}$ be the sequence generated by ADMM with $k < k_{\max_1}$, then the solution at a given step $1 \leq \hat{k} < k_{\max_1}$

is an approximate solution to the original problem with a bounded relative error stated in (54).

PROOF In the first stage, the first-order optimality condition of (45b) at the k th step can be written as $\nabla_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) = \mathbf{0}$. Then, let $\delta^{k+1} = \nabla_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) - \nabla_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^{k+1})$, and we have

$$\begin{aligned} & \|\delta^{k+1}\| \\ &= \left\| \nabla_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) - \nabla_{\mathbf{z}} \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^{k+1}) \right\| \\ &= \left\| (\mathbf{v}^k - \mathbf{v}^{k+1}) + (\boldsymbol{\mu}^k - \boldsymbol{\mu}^{k+1}) \mathbf{A}_{\mathcal{E}} \mathbf{x}^{k+1} \right. \\ &\quad \left. + (\boldsymbol{\Gamma}_z \circ (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k+1})) \mathbf{B}_{\mathcal{I}} \mathbf{x}^{k+1} \right\| \\ &\leq \left\| \mathbf{v}^k - \mathbf{v}^{k+1} \right\| + \left\| (\boldsymbol{\mu}^k - \boldsymbol{\mu}^{k+1}) \mathbf{A}_{\mathcal{E}} \mathbf{x}^{k+1} \right\| \\ &\quad + \left\| (\boldsymbol{\Gamma}_z \circ (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k+1})) \mathbf{B}_{\mathcal{I}} \mathbf{x}^{k+1} \right\| \\ &\leq c \left(\left\| \mathbf{v}^k - \mathbf{v}^{k+1} \right\| + \left\| (\boldsymbol{\mu}^k - \boldsymbol{\mu}^{k+1}) \right\| \right. \\ &\quad \left. + \left\| \boldsymbol{\Gamma}_z \circ (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k+1}) \right\| \right) \end{aligned} \quad (55)$$

where the first inequality holds due to the triangle inequality of norm, and the last inequality holds by taking $c = \max\{1, \|\mathbf{A}_{\mathcal{E}} \mathbf{x}^{k+1}\|, \|\mathbf{B}_{\mathcal{I}} \mathbf{x}^{k+1}\|\}$. Let $k = \hat{k}$ to complete the proof.

Theorem V.4 guarantees that when $k < k_{\max_1}$, the solutions from Algorithm (1) satisfy the bounded conditions. Next, we present the convergence proof for Algorithm (1) after applying the consensus updates in (45c).

LEMMA V.5 Let the conditions in Assumption V.2 hold, and let $\{\mathbf{x}^k, \mathbf{z}^k\}$ and $\{\mathbf{x}^{k+1}, \mathbf{z}^{k+1}\}$ be the sequences generated by Algorithm 1 in the second stage. Then, at the k th iteration of primal updates, we have

$$\begin{aligned} & \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^k, \mathbf{z}^k, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) \\ &\geq \frac{\kappa_x^k}{2} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 + \frac{\kappa_z^k}{2} \|\mathbf{z}^k - \mathbf{z}^{k+1}\|^2 \end{aligned} \quad (56)$$

where the constants $\kappa_x^k > \zeta_1^k > 0$ and $\kappa_z^k > \zeta_1^k > 0$ are independent of Λ^k .

PROOF At the k th iteration, for the \mathbf{x} -update in (45a) and \mathbf{z} -update in (45b), the Hessian matrices of the augmented Lagrangian are expressed as

$$\begin{aligned} \nabla_{\mathbf{x}}^2 \mathcal{L}(\mathbf{x}^{k+1}, \mathbf{z}^k, \Lambda^k) &= \zeta_1^k \mathbf{I} + \sum_{i \in \mathcal{E}} \zeta_2^k (\mathbf{A}_i \mathbf{z}^k) (\mathbf{A}_i \mathbf{z}^k)^T \\ &\quad + \zeta_3^k \sum_{j \in \mathcal{I}} \Gamma_{xj}^k (\mathbf{B}_j \mathbf{z}^k) (\mathbf{B}_j \mathbf{z}^k)^T \end{aligned} \quad (57a)$$

$$\begin{aligned} \nabla_{\mathbf{z}}^2 \mathcal{L}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) &= \zeta_1^k \mathbf{I} + \sum_{i \in \mathcal{E}} \zeta_2^k (\mathbf{A}_i \mathbf{x}^{k+1}) (\mathbf{A}_i \mathbf{x}^{k+1})^T \\ &\quad + \zeta_3^k \sum_{j \in \mathcal{I}} \Gamma_{zj}^k (\mathbf{B}_j \mathbf{x}^{k+1}) (\mathbf{B}_j \mathbf{x}^{k+1})^T. \end{aligned} \quad (57b)$$

With $[\zeta_1^k, \zeta_2^k, \zeta_3^k]$ being nondecreasing positive sequences, there exist positive constants κ_x and κ_z such that $\nabla_{\mathbf{x}}^2 \mathcal{L} \succeq$

$\kappa_x \mathbf{I} \succeq \xi_1 \mathbf{I}$ and $\nabla_{\mathbf{z}}^2 \mathcal{L} \succeq \kappa_z \mathbf{I} \succeq \xi_2 \mathbf{I}$ hold for all $\mathbf{x}, \mathbf{z} \in \mathbb{R}^n$. Meanwhile, the positive definiteness of \mathbf{Q}_x^k and \mathbf{Q}_z^k ensures the feasibility of these subproblems. Therefore, there exist the following inequalities for the \mathbf{x} -update in (45a) with a given pair $(\mathbf{z}^k, \Lambda^k)$ at the k th step:

$$\begin{aligned} & \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^k, \mathbf{z}^k, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^k, \Lambda^k) \\ & \geq \langle \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^{k+1}, \mathbf{z}^k, \Lambda^k), (\mathbf{x}^k - \mathbf{x}^{k+1}) \rangle \\ & + \frac{\kappa_x^k}{2} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \\ & = \frac{\kappa_x^k}{2} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \geq \frac{\xi_1^k}{2} \|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \end{aligned} \quad (58)$$

where the first inequality holds due to the strong convexity of $\mathcal{L}_{\mathbf{p}^k}(\mathbf{x}, \mathbf{z}^k, \Lambda^k)$ with respect to \mathbf{x} , and the equality holds due to the first-order optimality condition of (45a) with respect to \mathbf{x}^{k+1} . Similarly, the following inequality holds for the \mathbf{z} -update in (45b) with a given pair $(\mathbf{x}^{k+1}, \Lambda^k)$ at the k th step,

$$\begin{aligned} & \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \hat{\mathbf{z}}^k, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) \\ & \geq \frac{\kappa_z^k}{2} \|\hat{\mathbf{z}}^k - \mathbf{z}^{k+1}\|^2. \end{aligned} \quad (59)$$

Then, by adding (58) and (59), we obtain the condition in (56), which completes the proof.

To distinguish \mathbf{x}^{k+1} and \mathbf{z}^{k+1} in (45a) and (45b) from those obtained after applying the consensus updates in (45c), we denote the average solution after applying (45c) as

$$\hat{\mathbf{x}}^{k+1} = \hat{\mathbf{z}}^{k+1} = \frac{\mathbf{x}^{k+1} + \mathbf{z}^{k+1}}{2}. \quad (60)$$

Let d^* denote the dual optimal value, and $\{\mathbf{x}^k, \mathbf{z}^k\}$ and $\{\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k\}$ be the sequences generated by Algorithm 1 in the second stage optimization. Denote a dual function at the k th step as $d(\Lambda) = \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda)$, which is almost everywhere differentiable with respect to Λ . Moreover, for given $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$, we have

$$\nabla_{\mu} d(\Lambda) = \hat{\mathbf{x}}^T \mathbf{A}_{\mathcal{E}} \hat{\mathbf{z}} - \mathbf{a} \quad (61a)$$

$$\nabla_{\lambda} d(\Lambda) = \mathbf{\Gamma} \circ (\hat{\mathbf{x}}^T \mathbf{B}_{\mathcal{I}} \hat{\mathbf{z}} - \mathbf{b}). \quad (61b)$$

Then, the gap between the dual optimality d^* and the dual solution $d(\Lambda^k)$ at the k th iteration can be defined as $\Delta_d^k := d^* - d(\Lambda^k) = d^* - \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k)$. In addition, the primal gap between the primal optimality and the dual solution at the k th iteration is defined as $\Delta_p^k = \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - d(\Lambda^k)$.

To establish the linear convergence of Algorithm 1, we will introduce several lemmas for the bounded primal and dual gaps.

LEMMA V.6 Let assumptions V.2 and V.3 hold, then there exist positive scalars r_1^k and r_2^k such that

$$\Delta_d^k \leq r_1^k \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_{\mathcal{E}} \hat{\mathbf{z}}^k - \mathbf{a}\|^2 + r_2^k \|\mathbf{\Gamma}_{\Lambda}^k \circ ((\hat{\mathbf{x}}^k)^T \mathbf{B}_{\mathcal{I}} \hat{\mathbf{z}}^k - \mathbf{b})\|^2$$

where $\mathbf{\Gamma}_{\Lambda}^k = \mathbf{\Gamma}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k)$. Moreover, there exist positive scalars l_1^k and l_2^k independent of Λ^k such that

$$\Delta_p^k \leq l_1^k \|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\|^2 + l_2^k \|\hat{\mathbf{z}}^k - \mathbf{z}^{k+1}\|^2. \quad (62)$$

PROOF This lemma demonstrates the primal gap and dual gap at each iteration should be upper bounded. The details of the proof can be found in Appendix B.

Based on the results in Lemma V.6, we further consider the descent of primal and dual gaps at two adjacent iterations as follows.

LEMMA V.7 Let assumptions V.2 and V.3 hold, then for each $k \geq 1$, there exists α_1^k and α_2^k such that

$$\begin{aligned} \Delta_d^k - \Delta_d^{k-1} & \leq -\frac{\xi_2^{k-1} + \xi_2^k}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_{\mathcal{E}} \hat{\mathbf{z}}^k - \mathbf{a}\|^2 \\ & - \frac{\xi_3^{k-1} \|\mathbf{\Gamma}_{\Lambda}^{k-1}\| - \xi_3^k \|\mathbf{\Gamma}_x^k\|}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{B}_{\mathcal{I}} \hat{\mathbf{z}}^k - \mathbf{b}\|^2 \end{aligned} \quad (63a)$$

$$\begin{aligned} \Delta_p^k - \Delta_p^{k-1} & \leq -l_1^k \|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\|^2 - l_2^k \|\hat{\mathbf{z}}^k - \mathbf{z}^{k+1}\|^2 \\ & + \alpha_1^k \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_{\mathcal{E}} \hat{\mathbf{z}}^k - \mathbf{a}\|^2 + \alpha_2^k \|(\hat{\mathbf{x}}^k)^T \mathbf{B}_{\mathcal{I}} \hat{\mathbf{z}}^k - \mathbf{b}\|^2. \end{aligned} \quad (63b)$$

PROOF See Appendix C. ■

Then, the bounded primal and dual descent gaps can be used to show the linear convergence rate result for the hybrid ADMM.

THEOREM V.8 Let Assumptions V.2 and V.3 hold, and all elements of $\{\mathbf{p}^k\}$ be nondecreasing positive sequences. Then, the sequence of iterates $\{\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k\}$ generated from Algorithm 1 in the second stage converges linearly to an optimal primal-dual solution for the problem (41).

PROOF The proof for Theorem V.8 can be found in Appendix D. ■

VI. SIMULATION

In this section, we present simulation results of the dual-quaternion-based entry trajectory optimization problem solved by the hybrid ADMM algorithm. To verify the improved computational performance, we also provide the comparative results using the traditional entry dynamical model based on flight-path coordinates and a commercial NLP solver [46]. Another NLP solver, GPOPS II [47] has been applied to the same problem. However, it was not able to find a convergent solution. All simulation cases from the hybrid ADMM were run in MATLAB environments on a 3.6-GHz Desktop with 32-GB RAM. The commercial NLP solver has a MATLAB interface for setting the problem inputs and optimized codes are generated to improve computational efficiency, e.g., matrix sparsification, differential algebraic equations, and direct calling C++/C subroutines.

A. Problem Settings

The simulation presented here uses the hypersonic inflatable aerodynamic decelerator (HIAD) in [9] as the entry vehicle. HIAD has a symmetric shape without back shell covers. The payload is protected by a large diameter heat shield. Thus, the angle of attack and side-slip angle are constrained within $-20^\circ \leq \alpha \leq 20^\circ$ and $-20^\circ \leq \beta \leq 20^\circ$ to avoid flow impingement and large radiative heating effects on the payload at a high angle of attack. In addition, the lift

TABLE I
Parameters of the Entry Vehicle

Symbol	Value	Unit
m	51,099	kg
S	$\pi \frac{20^2}{4}$	m^2
\mathbf{J}	$\text{diag}(4.3799, 3.9857, 3.9857) \times 10^5$	$\text{kg} \cdot \text{m}^2$
c_{d_0}	1.556	—
c_{d_1}	-0.004114	—
c_{d_2}	-1.182	—
c_{l_0}	-0.003636	—
c_{l_1}	-0.7813	—
c_{y_0}	0	—
c_{y_1}	-0.7813	—

TABLE II
Mission Settings and Boundary Conditions

Symbol	Value	Unit
k_Q	$1.9027 \times 10^{-8} \times (\sqrt{R_m g_0})^{3.15}$	—
Q_{\max}	800	W/cm^2
R_f	$R_m + 7$	km
Ω	7.0808×10^{-5}	rad/s
n_{\max}	14.5	—
g_0	3.7114	m/s^2
M_x^L, M_y^L, M_z^L	-10	$\text{N} \cdot \text{m}$
M_x^U, M_y^U, M_z^U	10	$\text{N} \cdot \text{m}$

TABLE III
Initial States of the Entry Phase

Symbol	Value	Unit
$h(t_0)$	100	km
$V(t_0)$	4700	m/s
$\theta(t_0)$	120	deg
$\phi(t_0)$	40	deg
$\gamma(t_0)$	12.9	deg
$\psi(t_0)$	0	deg
$\sigma(t_0)$	-30	deg
$\alpha(t_0)$	17	deg
$\beta(t_0)$	-10	deg
$\omega_x(t_0), \omega_y(t_0), \omega_z(t_0)$	0	rad/s

and drag coefficients of the vehicle, C_L and C_D , are obtained from [48]. The parameters of the entry vehicle are listed in Table I. In addition, the mission settings and boundary conditions are listed in Table II.

Meanwhile, the initial states of the entry vehicle are given in Table III.

In Table III, ω_x , ω_y , and ω_z are the angular velocity terms along the body x -, y -, z -axes, respectively. Note that the attitude angles, including roll, pitch, and yaw, can be determined from α , β , γ , ψ , and σ . Then, the initial states of dual quaternions $\tilde{\mathbf{q}}(t_0) = [\hat{\mathbf{q}}_r(t_0), \hat{\mathbf{q}}_d(t_0)]^T$ can be obtained via (7), (22), (30), and (31), and $\hat{\mathbf{w}}(t_0) = [\hat{\omega}(t_0), \hat{\mathbf{v}}_M(t_0)]^T \in \mathbb{Q}$, $\hat{\mathbf{v}}_I$ can be determined by (16), (24), (25), and (26) accordingly.

B. Simulation Results

In this section, the simulation results using the dual-quaternion-based model combined with the hybrid ADMM algorithm and the traditional model combined with a commercial NLP solver are presented. The precision of both methods are compared by integrating the corresponding dynamics forward using the obtained control moments to find the terminal altitude and velocity and corresponding errors. In addition, robustness of algorithm convergence and cost values are compared for both methods. The entry trajectory is discretized into 31 nodes for both methods. For the dual-quaternion-based entry trajectory optimization problem, the resulting QCQP formulation includes 3070 unknown variables, associated with 3257 equality constraints and 558 inequality constraints. In addition, the initial guess \mathbf{x}^0 used for the first stage of the hybrid ADMM is generated by giving a linear control \mathbf{M} varying from the lower bounds \mathbf{M}^L to the upper bounds \mathbf{M}^U , and then, integrating the dynamics forward till the altitude reaches the specified altitude R_f . The same initial guess is set for the NLP method.

According to Proposition III.1, the optimal moment components should be bang–bang profiles. Then, for the initialization of the second stage in the hybrid ADMM, the obtained controls from the first stage will be rounded to bang–bang profiles by rounding positives to the upper bound and negatives to the lower bound, as shown in Fig. 2, where solutions from each iteration in the first stage are also provided. Without considering the bang–bang constraint in the first stage, it allows searching for a close to bang-bang control profile in a larger searching space with more flexibility. To improve the accuracy of the discretized trajectory in the second stage, the time intervals before and after the switching points in the bang–bang profiles are considered as additional variables . In addition, to illustrate the accelerated convergence rate in the second stage, the value of constraints error $\|\xi\|_1$ along the iterations of the two stages are shown in Fig. 3, where the second stage demonstrates the linearly reducing $\|\xi\|_1$.

The time histories of control and state variables from both methods are shown in Fig. 4. The duration of the entry phase from the ADMM is 213.5 s, which is shorter than the duration, 218.1 s, obtained from NLP. As shown in Fig. 4, the control commands generated from the ADMM have exact bang–bang control profiles, while the controls from the NLP, e.g., M_y , are not exact bang–bang curves. Compared to ADMM, NLP results may generate undesired jitters, which is impractical for implementing an irregular control inputs when it happens. To improve the precision of the NLP solutions, we gradually increase the number of discrete nodes. The maximum number of discrete notes for the NLP that leads to a convergent solution is 61. Further increasing the number of discrete nodes cannot obtain a convergent solution. However, the bang–bang control profile with 61 nodes generates more undesired jitters, as shown in Fig. 4. Therefore, the following comparison between the two methods uses the same 31 nodes.

Fig. 4(d) presents the time histories of α and β , where the two algorithms yield similar curves. Similar observations are found for the time history of the bank angle in Fig. 4(e), as well as the time histories of pitch, yaw, roll angles in

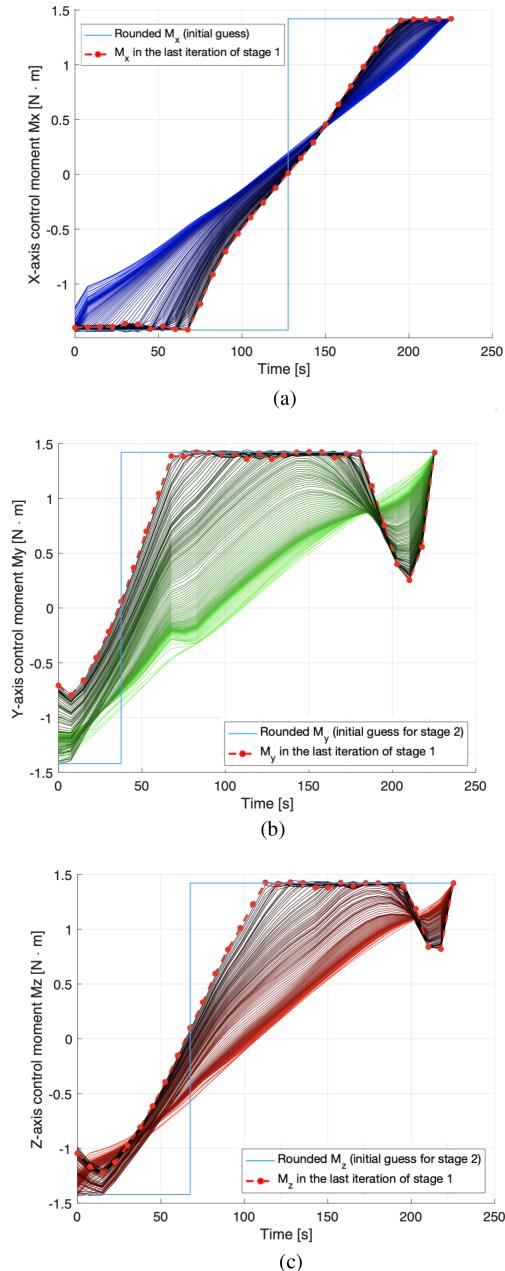


Fig. 2. Moment components from iterations of Stage 1 and final rounded value. (a) M_x values. (b) M_y values. (c) M_z values.

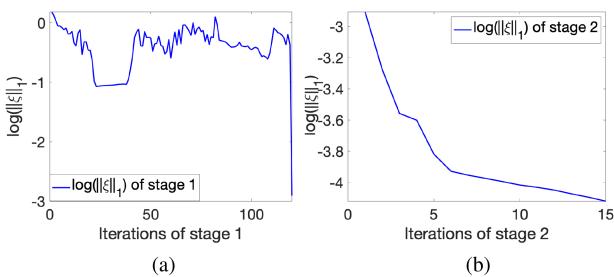


Fig. 3. Constraints error along iterations of two stages. (a) Constraints error along Stage 1. (b) Constraints error along Stage 2.

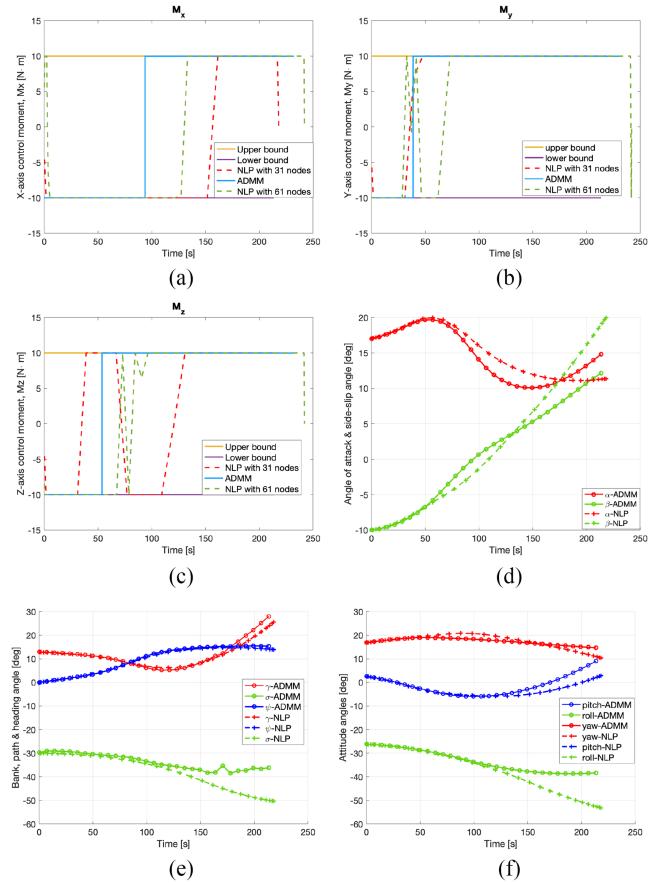


Fig. 4. Time histories of control and state variables from ADMM and NLP solutions. (a) M_x time histories. (b) M_y time histories. (c) M_z time histories. (d) α and β angles. (e) Bank, path, and heading angles. (f) Roll, pitch, and yaw angles.

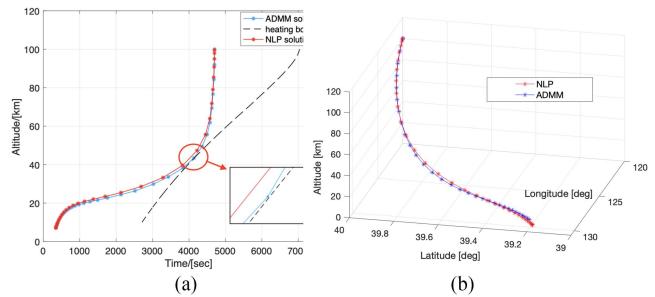


Fig. 5. Solution comparison from ADMM and NLP. (a) Altitude versus velocity. (b) Optimized 3-D trajectories.

Fig. 4(f). These plots all start with the same trends, and then, have small differences at the end. Fig. 5(a) presents the altitude versus velocity curves, where the red-star curve represents the solution from NLP, and the blue-star curve represents the ADMM solution. The final velocity of the NLP solution is 363.7 m/s at the altitude of 7.05 km. However, the terminal velocity of the ADMM solution is 346.6 m/s at the altitude of 7.00 km, which is smaller than the final velocity obtained from the NLP. Fig. 5(b) shows the 3-D trajectories of the NLP and the ADMM solutions. Through comparison of both methods, it indicates that the results from the ADMM yield an exact bang–bang control profile with a smaller terminal speed, which is feasible for

TABLE IV
Comparison of the Terminal Altitude and Velocity Errors, Terminal Speed, and Computational Time for ADMM and NLP

Method	Altitude Error	Relative altitude error	Speed Error	Relative Speed error	Terminal Speed	Computational Time
ADMM	52.0 m	0.74%	1.3 m/s	0.38%	346.3 m/s	189.0 s
NLP	551.7 m	7.88%	8.8 m/s	2.42%	363.7 m/s	12.1 s

implementation and benefits the following powered descent phase due to the reduced terminal speed.

To quantify the precision, two indexes are introduced. One is the terminal altitude error that is the error between the terminal altitude in the discretized solution and the one from integrating the entry dynamics using the optimized control commands. The other index is the terminal velocity error that represents the error between the terminal velocity in the discretized solution and the one from integration of dynamics using the optimized control solutions. The integration employs 2000 uniformly spaced points to interpolate the discrete solutions from the NLP method. These errors, together with the terminal speed and computational time, for both methods are listed in Table IV. When using the same number of discrete nodes, the solution from the ADMM using the dual-quaternion-based model leads to much reduced errors at the terminal point compared to those obtained from the NLP and the traditional dynamical model. The ADMM requires more computational time in the MATLAB environments. However, according to the computational performance comparison records, the commercial NLP solver with optimized codes is generally an order of magnitude times faster than MATLAB programs [46]. Furthermore, each iteration of the ADMM is a closed-form update based on simple linear vector/matrix operations, which does not require any optimization solver and is highly implementable for real-time computations. The most time-consuming step in each closed-form update is the inverse operation of large-scale matrices. Our future work will apply more efficient algorithms for computing inverse of large-scale matrices.

Moreover, to examine the robustness of the proposed method, simulations with varying initial conditions are conducted for both the ADMM and the NLP. 500 random cases were generated for robustness analysis. These 500 random cases have different initial altitude h_0 , initial velocity V_0 , and flight-path angle γ_0 within the altitude range $100 \text{ km} \leq h(t_0) \leq 110 \text{ km}$, velocity range $4700 \text{ m/s} \leq V(t_0) \leq 4800 \text{ m/s}$, and flight-path angle range $12.8^\circ \leq \gamma(t_0) \leq 13.0^\circ$. The 500 cases are generated around the nominal initial altitude $h_0 = 100 \text{ km}$, initial velocity $V_0 = 4700 \text{ m/s}$, and initial flight path angle $\gamma_0 = 12.9^\circ$ with disturbances, set as $h(t_0) = h_0 + \delta h$, $V(t_0) = V_0 + \delta V$, and $\gamma(t_0) = \gamma_0 + \delta \gamma$, where δh , δV , and $\delta \gamma$ follow a uniform distribution with $\delta h \sim U[0, 10] \text{ km}$, $\delta V \sim U[0, 100] \text{ m/s}$, and $\delta \gamma \sim U[0, 0.2] \text{ deg}$. All of these cases have a converged solution using the ADMM, while 236 out of 500 cases fail to converge using the NLP. The terminal altitude errors and terminal velocity errors for converged solutions from both methods are shown in Fig. 6(a), where 99% of the solutions from the ADMM lead to terminal altitude errors smaller than 200 m and terminal velocity errors smaller

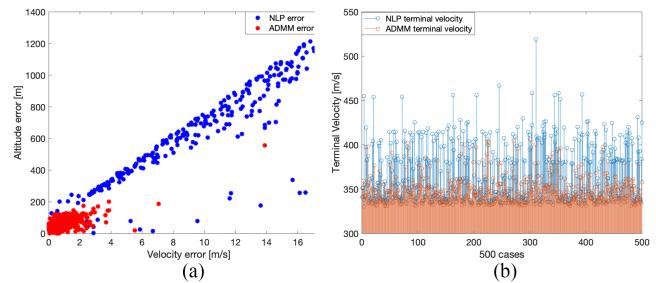


Fig. 6. Error and terminal velocity comparison of NLP and ADMM for the 500 cases. (a) Altitude and velocity errors. (b) Terminal velocity comparison.

than 4 m/s. Compared to the ADMM, almost all of the 264 converged solutions obtained from the NLP have terminal altitude errors larger than 200 m and 90% of the cases have the terminal velocity error larger than 4 m/s. Moreover, we compare the terminal speed for all cases integrated with the controls solved by the ADMM and the NLP, shown in Fig. 6(b), where 361 out of 500 cases obtain smaller terminal speeds for the solutions from the ADMM than those obtained from the NLP solver.

From the aforementioned comparison, when solving the 6-DoF entry trajectory optimization problem, it can be observed that the hybrid ADMM has advantages over the commercial NLP solver in terms of robustness, precision, and cost value. Moreover, the ADMM guarantees the exact bang–bang control profile. The commercial NLP solver with optimized codes has the advantage of reduced computational time compared to the ADMM run in MATLAB environments.

VII. CONCLUSION

This article examines the 6-DoF entry trajectory optimization problem that considers both translation and rotation motions. Instead of using the flight-path coordinates-based dynamics, the unit dual quaternion is employed to represent rigid body dynamics to reduce the nonlinearity and avoid the singularity of the rotational matrix. Then, combining the dual-quaternion-based dynamics and constraints on control and states, the 6-DoF entry trajectory optimization problem is formulated as a polynomial optimal control problem, which is then equivalently converted into a nonconvex QCQP problem. A hybrid ADMM is developed to solve the resulting QCQP with a guaranteed bounded error and a linear convergence rate. Comparative simulation results are provided to verify the advantages of the dual-quaternion-based model and the hybrid ADMM algorithm in terms of precision, implementability, and optimality when solving the 6-DoF entry trajectory optimization problem.

APPENDIX A PROOF OF PROPOSITION III.1

PROOF For the dual-quaternion-based 6-DoF entry trajectory optimization problem in (33), the Hamiltonian function can be expressed as $H = \lambda_J^T \|\mathbf{A}\tilde{\mathbf{w}}(t_f)\| + \frac{1}{2}\lambda_q^T(\tilde{\mathbf{q}} \otimes \tilde{\mathbf{w}}) + \lambda_w^T J_d^{-1}(\Phi\tilde{\mathbf{F}}_B + \tilde{\mathbf{G}}_B + \tilde{\mathbf{M}}_B - \tilde{\mathbf{w}} \otimes \mathbf{J}_d \tilde{\mathbf{w}}) + \lambda_Q(\dot{Q} - Q_{\max}) + \lambda_n(\|\mathbf{F}\| - n_{\max})$, where λ_J , λ_q , λ_w , λ_Q , and λ_n are the Lagrange multipliers associated with the dynamics and the inequality constraints. Since the control moment components $\mathbf{M} = [M_x, M_y, M_z]$ appears as linear terms in the Hamiltonian function, the first-order necessary conditions of optimality does not include the moment components, which indicates that the optimal moment components have bang–bang profiles [49].

APPENDIX B PROOF OF LEMMA V.6

PROOF Let Λ^* denote an optimal dual solution of the problem (41). Based on the mean value theorem, there exists some $\tilde{\Lambda}$ in the line segment joining Λ^k and Λ^* such that

$$\begin{aligned} \Delta_d^k &= d(\Lambda^*) - d(\Lambda^k) = \langle \nabla d(\tilde{\Lambda}), \Lambda^* - \Lambda^k \rangle \\ &= \langle \nabla d(\tilde{\Lambda}) - \nabla d(\Lambda^k), \Lambda^* - \Lambda^k \rangle \\ &\leq \|\nabla_\mu d(\tilde{\Lambda}) - \nabla_\mu d(\Lambda^k)\| \cdot \|\mu^* - \mu^k\| \\ &\quad + \|\nabla_\lambda d(\tilde{\Lambda}) - \nabla_\lambda d(\Lambda^k)\| \cdot \|\Gamma_\Lambda^* \circ \lambda^* - \Gamma_\Lambda^k \circ \lambda^k\| \\ &\leq \frac{1}{\rho_\mu} \|\tilde{\mu} - \mu^*\| \cdot \|\mu^k - \mu^*\| \\ &\quad + \frac{1}{\rho_\mu} \|\Gamma_\Lambda^* \circ \lambda^* - \Gamma_\Lambda^k \circ \lambda^k\|_F \cdot \|\Gamma_\Lambda^* \circ \lambda^* - \Gamma_\Lambda^k \circ \lambda^k\| \\ &\leq \frac{1}{\rho_\mu} \|\mu^k - \mu^*\|^2 + \frac{1}{\rho_\lambda} \|\Gamma_\Lambda^* \circ \lambda^* - \Gamma_\Lambda^k \circ \lambda^k\|^2 \\ &\leq r_1^k \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a}\|^2 + r_2^k \|\Gamma_\Lambda^k \circ ((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b})\|^2 \end{aligned}$$

where the second inequality holds due to the mean value theorem, the last inequality holds due to the Lipschitz continuity of $\nabla d(\Lambda)$ and $r_1^k = \frac{L_{1x}^k}{\rho_\mu}$, $r_2^k = \frac{L_{1z}^k}{\rho_\lambda}$. Next, we consider the primal gap Δ_p^k

$$\begin{aligned} \Delta_p^k &= \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k) \\ &= \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) \\ &\quad + \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k) \end{aligned} \quad (64)$$

where $\mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k)$ is bounded by

$$\begin{aligned} &\mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) \\ &\quad + \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^k}(\mathbf{x}^{k+1}, \mathbf{z}^{k+1}, \Lambda^k) \\ &\leq \frac{L_{1x}^k}{2} \|\mathbf{x}^{k+1} - \hat{\mathbf{x}}^{k+1}\|^2 + \frac{L_{1z}^k}{2} \|\mathbf{z}^{k+1} - \hat{\mathbf{z}}^{k+1}\|^2 \\ &\leq \frac{L_{1x}^k + L_{1z}^k}{8} (\|\mathbf{x}^{k+1} - \hat{\mathbf{x}}^k\|^2 + \|\mathbf{z}^{k+1} - \hat{\mathbf{z}}^k\|^2) \end{aligned} \quad (65)$$

where the first inequality is due to the Lipschitz continuity of $\mathcal{L}(\mathbf{x}, \mathbf{z}, \Lambda)$ with given \mathbf{x} or \mathbf{z} , L_{1x}^k and L_{1z}^k are the Lipschitz

constants, while the second inequality follows from the facts that $\hat{\mathbf{x}}^k = \hat{\mathbf{z}}^k$ and the triangle inequality for norms. Then, combining (56) and (65), we can set the bound of Δ_p^k as (62) by denoting

$$l_1^k = \frac{L_{1x}^k + L_{1z}^k}{8} - \kappa_x^k, \quad l_2^k = \frac{L_{1x}^k + L_{1z}^k}{8} - \kappa_z^k. \quad (66)$$

APPENDIX C PROOF OF LEMMA V.7

PROOF The difference between two adjacent optimality gaps can be bounded as follows:

$$\begin{aligned} \Delta_d^k - \Delta_d^{k-1} &= [d^* - d(\Lambda^k)] - [d^* - d(\Lambda^{k-1})] \\ &= \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^{k-1}, \hat{\mathbf{z}}^{k-1}, \Lambda^{k-1}) - \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^{k-1}) \\ &\quad + \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^{k-1}) - \mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k) \\ &= \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^{k-1}, \hat{\mathbf{z}}^{k-1}, \Lambda^{k-1}) - \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^{k-1}) \\ &\quad + \langle \mu^{k-1} - \mu^k, (\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a} \rangle \\ &\quad + \frac{\zeta_2^{k-1} - \zeta_2^k}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a}\|^2 \\ &\quad + \langle \Gamma_\Lambda^{k-1} \circ \lambda^{k-1} - \Gamma_x^k \circ \lambda^k, (\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b} \rangle \\ &\quad + \frac{(\zeta_3^{k-1} \Gamma_\Lambda^{k-1} - \zeta_3^k \Gamma_x^k)}{2} \circ \|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b})\|^2 \\ &\leq -\frac{\zeta_2^{k-1} + \zeta_2^k}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a}\|^2 \\ &\quad - \frac{\zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\| - \zeta_3^k \|\Gamma_x^k\|}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b}\|^2 \end{aligned} \quad (67)$$

where the last inequality holds due to the updates of the dual variables Λ^{k-1} and the primal descent at the k th iteration. Next, we proceed to bound the decrease in the primal gap Δ_p^k . Based on the definition of the augmented Lagrangian function in (44) and the updating rule of Λ , we have $\mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k) = \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^{k-1}) + \frac{3\zeta_2^k - \zeta_2^{k-1}}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a}\|^2 + \frac{(3\zeta_3^k \Gamma_x^k - \zeta_3^{k-1} \Gamma_\Lambda^{k-1})}{2} \circ \|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b})\|^2$. Recall from Lemma V.6 that Algorithm 1 gives a sufficient descent of Δ_p^k . Hence, we have the following bound on the decrease of primal gap

$$\begin{aligned} \Delta_p^k - \Delta_p^{k-1} &= [\mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^{k+1}, \hat{\mathbf{z}}^{k+1}, \Lambda^k) - \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k)] \\ &\quad - [\mathcal{L}_{\mathbf{p}^k}(\hat{\mathbf{x}}^k, \hat{\mathbf{z}}^k, \Lambda^k) - \mathcal{L}_{\mathbf{p}^{k-1}}(\hat{\mathbf{x}}^{k-1}, \hat{\mathbf{z}}^{k-1}, \Lambda^{k-1})] \\ &\leq (\zeta_2^k - \zeta_2^{k-1}) \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{z}}^k - \mathbf{a}\|^2 \\ &\quad + (\zeta_3^k \|\Gamma_x^k\| - \zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\|) \circ \|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{z}}^k - \mathbf{b})\|^2 \\ &\quad - \kappa_x^k \|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\|^2 - \kappa_z^k \|\hat{\mathbf{z}}^k - \mathbf{z}^{k+1}\|^2. \end{aligned}$$

Then, the proof is completed by denoting $\alpha_1^k = \zeta_2^k - \zeta_2^{k-1}$ and $\alpha_2^k = \zeta_3^k \|\Gamma_x^k\| - \zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\|$.

APPENDIX D PROOF OF THEOREM V.8

To prove Theorem V.8, we consider the sum of optimality gaps $\Delta_p^k + \Delta_d^k$. First, we combine the two estimates in (63a) and (63b)

$$\begin{aligned} [\Delta_p^k + \Delta_d^k] - [\Delta_p^{k-1} + \Delta_d^{k-1}] &= [\Delta_p^k - \Delta_p^{k-1}] + [\Delta_d^k - \Delta_d^{k-1}] \\ &\leq \frac{\zeta_2^k - 3\zeta_2^{k-1}}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{y}}^k - \mathbf{a}\|^2 \\ &\quad + \frac{\zeta_3^k \|\Gamma_x^k\| - 3\zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\|}{2} \|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{y}}^k - \mathbf{b})\|^2 \\ &\quad - \kappa_x^k \|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\|^2 - \kappa_z^k \|\hat{\mathbf{y}}^k - \mathbf{y}^{k+1}\|^2. \end{aligned} \quad (68)$$

Obviously, if the penalty coefficients $\{\zeta_1^k, \zeta_2^k, \zeta_3^k\}$ satisfy $\zeta_2^k - 3\zeta_2^{k-1} \leq 0$ and $\zeta_3^k \|\Gamma_x^k\| - 3\zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\| \leq 0$, the estimate in (68) shows that $[\Delta_p^k + \Delta_d^k] \leq [\Delta_p^{k-1} + \Delta_d^{k-1}]$. Moreover, the descent condition in (68) holds for all $k \geq 1$, and we have $\|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\| \rightarrow 0$, $\|\hat{\mathbf{y}}^k - \mathbf{y}^{k+1}\| \rightarrow 0$, $\|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{y}}^k - \mathbf{a}\| \rightarrow 0$, $\|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{y}}^k - \mathbf{b})\| \rightarrow 0$, which indicate the limit point of $\{\hat{\mathbf{x}}^k, \hat{\mathbf{y}}^k\}$ is an optimal solution of problem (41).

Next, we will show the linear convergence rate of the proposed algorithm 1. Since Δ_d^k and Δ_p^k are both nonnegative and upper bounded, it follows that $d(\Lambda^k)$ is bounded based on the following derivation:

$$\begin{aligned} &[\Delta_p^k + \Delta_d^k] - [\Delta_p^{k-1} + \Delta_d^{k-1}] \\ &\leq \frac{\zeta_2^k - 3\zeta_2^{k-1}}{2} \|(\hat{\mathbf{x}}^k)^T \mathbf{A}_\varepsilon \hat{\mathbf{y}}^k - \mathbf{a}\|^2 \\ &\quad + \frac{\zeta_3^k \|\Gamma_x^k\| - 3\zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\|}{2} \|((\hat{\mathbf{x}}^k)^T \mathbf{B}_I \hat{\mathbf{y}}^k - \mathbf{b})\|^2 \\ &\quad - l_1^k \|\hat{\mathbf{x}}^k - \mathbf{x}^{k+1}\|^2 - l_2^k \|\hat{\mathbf{y}}^k - \mathbf{y}^{k+1}\|^2 \\ &\leq - \min \left\{ \frac{\kappa_x^k}{l_1^k}, \frac{\kappa_z^k}{l_2^k} \right\} \Delta_d^k \\ &\quad - \min \left\{ \frac{3\zeta_2^{k-1} - \zeta_2^k}{2r_1^k}, \frac{3\zeta_3^{k-1} \|\Gamma_\Lambda^{k-1}\| - \zeta_3^k \|\Gamma_x^k\|}{2r_2^k} \right\} \Delta_d^k \\ &= \tau_d^k \Delta_d^k + \tau_p^k \Delta_p^k \end{aligned}$$

where the last inequality holds due to the upper bounded of Δ_d^k and Δ_p^k . Consequently, we have $[\Delta_p^k + \Delta_d^k] - [\Delta_p^{k-1} + \Delta_d^{k-1}] \leq -\tau^k [\Delta_p^k + \Delta_d^k]$, where τ^k is defined as $\tau^k = \min\{\tau_d^k, \tau_p^k\}$. This shows that the sequence $\{\Delta_d^k + \Delta_p^k\}$ converges to a stationary point R-linearly.¹

REFERENCES

- [1] J. Ridderhof, P. Tsiotras, and B. J. Johnson, "Stochastic entry guidance," *J. Guid., Control, Dyn.*, vol. 45, pp. 320–334, 2021.
- [2] N. Sarigul-Klijn et al., "Survey of planetary entry guidance algorithms," *Prog. Aerosp. Sci.*, vol. 68, pp. 64–74, 2014.
- [3] L. Cheng, F. Jiang, Z. Wang, and J. Li, "Multiconstrained real-time entry guidance using deep neural networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 1, pp. 325–340, Feb. 2021.
- [4] X. Liu, Z. Shen, and P. Lu, "Entry trajectory optimization by second-order cone programming," *J. Guid., Control, Dyn.*, vol. 39, no. 2, pp. 227–241, 2016.
- [5] L. Yang, X. Liu, W. Chen, and H. Zhou, "Autonomous entry guidance using linear pseudospectral model predictive control," *Aerospace Sci. Technol.*, vol. 80, pp. 38–55, 2018.
- [6] P. Lu, "Entry guidance: A unified method," *J. Guid., Control, Dyn.*, vol. 37, no. 3, pp. 713–728, 2014.
- [7] P. Lu, "Predictor-corrector entry guidance for low-lifting vehicles," *J. Guid., Control, Dyn.*, vol. 31, no. 4, pp. 1067–1075, 2008.
- [8] R. A. Lugo, R. Powell, and A. M. Dwyer-Cianciolo, "Overview of a generalized numerical predictor-corrector targeting guidance with application to human-scale Mars entry, descent, and landing," in *Proc. AIAA Scitech Forum*, 2020, Art. no. 0846.
- [9] A. Dwyer-Cianciolo and R. W. Powell, "Entry, descent, and landing guidance and control approaches to satisfy Mars human mission landing criteria," in *Proc. AAS/AIAA Space Flight Mechanics Conf.*, 2017, Art. no. 254.
- [10] M. Marwaha and J. Valasek, "Fault-tolerant control allocation for mars entry vehicle using adaptive control," *Int. J. Adapt. Control Signal Process.*, vol. 25, no. 2, pp. 95–113, 2011.
- [11] K. D. Mease, D. T. Chen, P. Teufel, and H. Schonenberger, "Reduced-order entry trajectory planning for acceleration guidance," *J. Guid., Control, Dyn.*, vol. 25, no. 2, pp. 257–266, 2002, doi: [10.2514/2.4906](https://doi.org/10.2514/2.4906).
- [12] P. Brugarolas, A. S. Martin, and E. Wong, "Attitude controller for the atmospheric entry of the mars science laboratory," in *Proc. AIAA Guid., Navigat. Control Conf. Exhibit.*, 2008, Art. no. 6812.
- [13] J. A. Leavitt and K. D. Mease, "Feasible trajectory generation for atmospheric entry guidance," *J. Guid., Control, Dyn.*, vol. 30, no. 2, pp. 473–481, 2007, doi: [10.2514/1.23034](https://doi.org/10.2514/1.23034).
- [14] C. Restrepo and J. Valasek, "Structured adaptive model inversion controller for Mars atmospheric flight," *J. Guid., Control, Dyn.*, vol. 31, no. 4, pp. 937–953, 2008.
- [15] H. J. Savino, L. C. Pimenta, J. A. Shah, and B. V. Adorno, "Pose consensus based on dual quaternion algebra with application to decentralized formation control of mobile manipulators," *J. Franklin Inst.*, vol. 357, no. 1, pp. 142–178, 2020.
- [16] M. Ratiu and M. A. Prichici, "Industrial robot trajectory optimization—A review," in *Proc. MATEC Web Conf.*, vol. 126, 2017, Art no. 02005.
- [17] Z. Wang and Z. Wu, "Six-DOF trajectory optimization for reusable launch vehicles via gauss pseudospectral method," *J. Syst. Eng. Electron.*, vol. 27, no. 2, pp. 434–441, 2016.
- [18] U. Lee and M. Mesbah, "Dual quaternions, rigid body mechanics, and powered-descent guidance," in *Proc. IEEE 51st IEEE Conf. Decis. Control*, 2012, pp. 3386–3391.
- [19] B. J. Johnson, R. R. Sostaric, and P. Lu, "Mid lift-to-drag ratio rigid vehicle 6-DOF performance for human Mars entry, descent, and landing: A fractional polynomial powered descent guidance approach," in *Proc. AIAA SciTech Forum*, 2020, Art. no. 1513.
- [20] R. Chai, A. Tsourdos, A. Savvaris, S. Chai, Y. Xia, and C. L. P. Chen, "Six-DOF spacecraft optimal trajectory planning and real-time attitude control: A deep neural network-based approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 5005–5013, Nov. 2020.
- [21] S. Li and Y. Peng, "Mars entry trajectory optimization using DOC and DCNLP," *Adv. Space Res.*, vol. 47, no. 3, pp. 440–452, 2011.
- [22] U. Lee and M. Mesbah, "Constrained autonomous precision landing via dual quaternions and model predictive control," *J. Guid. Control Dyn.*, vol. 40, no. 2, pp. 292–308, 2017.
- [23] J.-W. Kwon, D. Lee, and H. Bang, "Virtual trajectory augmented landing control based on dual quaternion for lunar lander," *J. Guid., Control, Dyn.*, vol. 39, no. 9, pp. 2044–2057, 2016.
- [24] J. C. Bedoya, A. Abdelhadi, C.-C. Liu, and A. Dubey, "A QCQP and SDP formulation of the optimal power flow including renewable energy resources," in *Proc. IEEE Int. Symp. Syst. Eng.*, 2019, pp. 1–8.

¹Suppose that $\{\mathbf{x}^k\}$ converges to $\bar{\mathbf{x}}$. The sequence is said to converge R-linearly to $\bar{\mathbf{x}}$ if $\|\mathbf{x}^k - \bar{\mathbf{x}}\| \leq c\tau^k$ for all k and for some $c > 0$.

- [25] A. Gharanjik, B. Shankar, M. Soltanalian, and B. Oftersten, "An iterative approach to nonconvex QCQP with applications in signal processing," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop*, 2016, pp. 1–5.
- [26] P. Biswas, T.-C. Lian, T.-C. Wang, and Y. Ye, "Semidefinite programming based algorithms for sensor network localization," *ACM Trans. Sensor Netw.*, vol. 2, no. 2, pp. 188–220, 2006.
- [27] H. Song and H.-C. Huang, "A successive convex approximation method for multistage workforce capacity planning problem with turnover," *Eur. J. Oper. Res.*, vol. 188, no. 1, pp. 29–48, 2008.
- [28] M. Razaviyayn, M. Hong, Z.-Q. Luo, and J.-S. Pang, "Parallel successive convex approximation for nonsmooth nonconvex optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1440–1448.
- [29] M. ApS, "MOSEK optimization toolbox for MATLAB," *User's Guide and Reference Manual, Version*, 2019, vol. 4.
- [30] M. Andersen et al., "Interior-point methods for large-scale cone programming," *Optim. Mach. Learn.*, vol. 5583, pp. 55–79, 2011.
- [31] Z. Wen, D. Goldfarb, and W. Yin, "Alternating direction augmented lagrangian methods for semidefinite programming," *Math. Program. Comput.*, vol. 2, no. 3–4, pp. 203–230, 2010.
- [32] P. Giselsson and S. Boyd, "Linear convergence and metric selection for Douglas-Rachford splitting and ADMM," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 532–544, Feb. 2017.
- [33] M. Ma, A. N. Nikolakopoulos, and G. B. Giannakis, "Hybrid ADMM: A unifying and fast approach to decentralized optimization," *EURASIP J. Adv. Signal Process.*, vol. 2018, no. 1, pp. 1–17, 2018.
- [34] Z. Xu, G. Taylor, H. Li, M. A. Figueiredo, X. Yuan, and T. Goldstein, "Adaptive consensus ADMM for distributed optimization," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3841–3850.
- [35] C. Sun, N. Kingry, and R. Dai, "A unified formulation and non-convex optimization method for mixed-type decision-making of robotic systems," *IEEE Trans. Robot.*, vol. 37, no. 3, pp. 831–846, Jun. 2021.
- [36] C. Sun, R. Dai, and P. Lu, "Multi-phase spacecraft mission optimization by quadratically constrained quadratic programming," in *Proc. AIAA Scitech Forum*, 2019, Art. no. 1667.
- [37] C. Wan, G. Jing, R. Dai, and J. R. Rea, "Fuel-optimal guidance for end-to-end human-Mars entry, powered-descent, and landing mission," in *Proc. AIAA Scitech Forum*, 2020, Art. no. 1472.
- [38] C. Wan, G. Jing, R. Dai, and J. Rea, "Fuel-optimal guidance for end-to-end human-Mars entry, powered-descent, and landing mission," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 4, pp. 2837–2854, Aug. 2022.
- [39] C. Wan, C. Pei, R. Dai, G. Jing, and J. R. Rea, "Six-dimensional atmosphere entry guidance based on dual quaternion," in *Proc. AIAA Scitech Forum*, 2021, Art. no. 0507.
- [40] J. M. McCarthy, *Introduction to Theoretical Kinematics*. Cambridge, MA, USA: MIT Press, 1990.
- [41] A. T. Yang, "Application of quaternion algebra and dual numbers to the analysis of spatial mechanisms," Ph.D. dissertation, Columbia Univ. Morningside Heights, New York, NY, USA, 1963.
- [42] A. Fedele, S. Carannante, M. Grassi, and R. Savino, "Aerodynamic control system for a deployable re-entry capsule," *Acta Astronautica*, vol. 181, pp. 707–716, 2021.
- [43] M. Sparapani, "Enabling mars exploration using inflatable Purdue aerodynamic decelerator with deployable entry systems (iPADDLES) technology," Ph.D. dissertation, Purdue Univ., West Lafayette, IN, USA, 2016.
- [44] C. Wan, R. Dai, and P. Lu, "Alternating minimization algorithm for polynomial optimal control problems," *J. Guid., Control, Dyn.*, vol. 42, no. 4, pp. 723–736, 2019.
- [45] S. Boyd, N. Parikh, and E. Chu, *Distributed Optimization and Statistical Learning Via the Alternating Direction Method of Multipliers*. Boston, MA, USA: Now Publishers Inc., 2011.
- [46] K. Holmström, A. O. Görän, and M. M. Edvall, *User's Guide for TOMLAB/PENOPT*, Tomlab Optimization Inc., Pullman, MA, USA, 2006.
- [47] M. A. Patterson and A. V. Rao, "GPOPS-II: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming," *ACM Trans. Math. Softw.*, vol. 41, no. 1, pp. 1–37, 2014.
- [48] B. P. Harper and R. D. Braun, "Preliminary design study of asymmetric hypersonic inflatable aerodynamic decelerators for mars entry," Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, AE8900 MS Special Problems Rep., 2014.
- [49] H. Maurer, C. Büskens, J.-H. Kim, and C. Kaya, "Optimization methods for the verification of second order sufficient conditions for bang-bang controls," *Optimal Control Appl. Methods*, vol. 26, no. 3, pp. 129–156, 2005.



Chaoying (Paige) Pei received the B.S. and M.S. degrees in instrumentation and optoelectronic engineering from Beihang University, Beijing, China, in 2015 and 2018, respectively. She is currently working toward the Ph.D. degree in aeronautics and astronautics engineering with Purdue University, West Lafayette, IN, USA.

Her research interests include control of autonomous systems and numerical optimization.



Changhuang (Charlie) Wan received the bachelor and master degrees in spacecraft design and engineering from Beihang University, Beijing, China, in 2009 and 2012, respectively, and the Ph.D. degree in aerospace engineering from The Ohio State University, Columbus, OH, USA, in 2021.

He is an Assistant Professor in aerospace science engineering with Tuskegee University, Tuskegee, AL, USA. His research interests include optimal control, numerical optimization,

and autonomous systems.



Ran Dai (Member, IEEE) received the B.S. degree in automation science from Beihang University, Beijing, China, in 2002, and the M.S. and Ph.D. degrees in aerospace engineering from Auburn University, Auburn, AL, USA, in 2015 and 2017, respectively.

She is an Associate Professor with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, USA. Her research interests include control of autonomous systems, numerical optimization, and networked dynamical systems.

Dr. Dai is an Associate Editor for *IEEE TRANSACTION ON AEROSPACE AND ELECTRONIC SYSTEMS*, and was the recipient of the National Science Foundation Career Award and NASA Early Faculty Career Award.



Jeremy R. Rea received the Bachelor of Science degree in aeronautical and astronautical engineering from The Ohio State University, Columbus, OH, USA, in 1999, the Master of Science degree in aeronautics and astronautics from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2001, and the Doctor of Philosophy degree in aerospace engineering from The University of Texas at Austin, Austin, TX, USA, in 2009, under a Johnson Space Center Fellowship.

He began work with the NASA Johnson Space Center, Houston, TX, USA, in August 2000, where he is currently the Orion Entry Guidance and Performance Subsystem Manager for Project Artemis and the Guidance Technical Discipline Lead.