Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Review:

Survey of autonomous guidance methods for powered planetary landing*

Zheng-yu SONG†‡[1,5], Cong WANG[2], Stephan THEIL[3], David SEELBINDER[3], Marco SAGLIANO[3], Xin-fu LIU[4], Zhi-jiang SHAO[5]

[1]*China Academy of Launch Vehicle Technology, Beijing 100076, China*
[2]*Beijing Aerospace Automatic Control Institute, Beijing 100854, China*
[3]*GNC Systems Department, DLR Institute of Space Systems, Bremen 28001, Germany*
[4]*School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China*
[5]*College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China*
†E-mail: zycalt12@sina.com

Received Aug. 31, 2019; Revision accepted Feb. 6, 2020; Crosschecked Mar. 1, 2020; Published online Mar. 14, 2020

Abstract: This paper summarizes the autonomous guidance methods (AGMs) for pinpoint soft landing on celestial surfaces. We first review the development of powered descent guidance methods, focusing on their contributions for dealing with constraints and enhancing computational efficiency. With the increasing demand for reusable launchers and more scientific returns from space exploration, pinpoint soft landing has become a basic requirement. Unlike the kilometer-level precision for previous activities, the position accuracy of future planetary landers is within tens of meters of a target respecting all constraints of velocity and attitude, which is a very difficult task and arouses renewed interest in AGMs. This paper states the generalized three- and six-degree-of-freedom optimization problems in the powered descent phase and compares the features of three typical scenarios, i.e., the lunar, Mars, and Earth landing. On this basis, the paper details the characteristics and adaptability of AGMs by comparing aspects of analytical guidance methods, numerical optimization algorithms, and learning-based methods, and discusses the convexification treatment and solution strategies for non-convex problems. Three key issues related to AGM application, including physical feasibility, model accuracy, and real-time performance, are presented afterward for discussion. Many space organizations, such as those in the United States, China, France, Germany, and Japan, have also developed free-flying demonstrators to carry out related research. The guidance methods which have been tested on these demonstrators are briefly introduced at the end of the paper.

Key words: Autonomous guidance method; Pinpoint soft landing; Powered descent; Nonlinear programming
https://doi.org/10.1631/FITEE.1900458 **CLC number:** V448; TP273

1 Introduction

Autonomous guidance methods (AGMs) plan flight trajectories and/or provide onboard guidance commands in real time to meet the complex constraints and terminal conditions in follow-up flight processes independent of off-line planned reference trajectories. An AGM considers the flight process as a whole, and has strong adaptability and robustness for tasks consisting of strong environmental uncertainties and strict constraints. In recent years, missions such as returning to the Moon, landing on Mars, and the recovery of launch vehicles have attracted worldwide attention. To obtain a greater

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (No. 61773341) and the International Academy of Astronautics Study Group SG 3.32
ORCID: Zheng-yu SONG, https://orcid.org/0000-0001-8011-4195

scientific return, pinpoint soft landing has been an indispensable feature, which sets the current research apart from those in the Apollo era. The landing processes of these missions are characterized by multiple flight phases and large uncertainties, and the actual flight trajectory may vary from the prescribed reference trajectory. Powered descent (PD) is the last phase of landing a flight vehicle, and the guidance, navigation, and control (GNC) system must be able to deal with accumulated errors from the earlier phases. For all the above missions, AGMs are considered as a key technology to achieve pinpoint landing.

The study of AGMs for powered landing began with the Apollo project. Due to the limitation of the computing ability in that era, the Apollo lunar module did not use a fuel-optimal algorithm based on optimal control theory (Meditch, 1964). Instead, it adopted the polynomial guidance method, which was easy to implement and deduced an analytical expression for the acceleration command in combination with the remaining flight time for soft landing (Klumpp, 1974). Restricted by the traditional variational method or Pontryagin maximum principle, it is very difficult to obtain analytical guidance laws for landing problems with complex constraints (for example, concurrently meeting the terminal constraints of velocity, position, and altitude, or dealing with the aerodynamic drag), and thus the progress in this area has remained slow.

During the early 21st century, the United States launched the Mars Exploration Program. The Mars Science Laboratory computed a three-dimensional polynomial trajectory online during the PD phase (Prakash et al., 2008), and AGMs for planetary exploration, especially for pinpoint landing, attracted renewed attention. Thanks to 40 years of theoretical mathematicians' research on numerical programming algorithms (Luenberger and Ye, 1984; Boggs and Tolle, 1995; Wright, 1997; Boyd and Vandenberghe, 2004; Bomze et al., 2007) and the evolution of hardware, many solvers have been developed (Toh et al., 2004; Gill et al., 2005; Grant et al., 2008; Biegler and Zavala, 2009; Domahidi et al., 2013). To improve a lander's autonomy during a pinpoint soft landing, researchers are no longer confined to explicit guidance laws based on simplified models. Instead, computational guidance that considers motion characteristics and process constraints has gradually become an important research focus (Lu, 2017; Tsiotras and Mesbahi, 2017). Through reasonable modeling, transformations, and analyses, we can rely on robust and efficient numerical algorithms and advanced computing platforms for iterative calculations to meet the requirements of online flight trajectory planning. These methods are regarded as the state-of-the-art technologies in pinpoint landing, but they are still restricted by computational efficiency and convergence, and the practical applications are rare.

In early studies, the lander was first taken as a mass particle, and the direct method was applied to solve the three-degree-of-freedom (3-DoF) nonlinear programming (NLP) problems with a fuel-optimality objective. By analyzing the motion and control of planetary powered pinpoint landings (Topcu et al., 2005, 2007; Najson and Mease, 2006), it was found that the pseudospectral discretization methods developed from the finite element theory had high solution accuracy in solving NLP problems (Sostaric and Rea, 2005; Fahroo and Ross, 2008), although their real-time performance and convergence were unable to meet the requirements of online trajectory planning. Açıkmeşe and Ploen (2005, 2007) proposed to transform the 3-DoF Mars landing problem into semi-definite programming (SDP) and second-order cone programming (SOCP), and Ploen et al. (2006) compared the features of polynomial guidance with those from the convex optimization algorithm. It was proved in follow-up studies that the convex problem, after lossless convexification, has the same solution to the original problem based on the Pontryagin maximum principle (Açıkmeşe and Blackmore, 2011; Blackmore et al., 2012; Açıkmeşe et al., 2013b; Harris and Açıkmeşe, 2014). The convergence and rapidity of the convex optimization algorithms make AGMs, which are based on numerical algorithms, have the potential for online applications. The Jet Propulsion Laboratory (JPL) developed the Guidance for Fuel-Optimal Large Diverts (G-FOLD) software in the Autonomous Ascent and Descent Powered-Flight Testbed (ADAPT), where the computation time can reach 100 ms on a 1.4-GHz processor. The real-time performance and control accuracy of the algorithm were verified with the Xombie vehicle (Açıkmeşe et al., 2013a; Dueri et al., 2014, 2017; Scharf et al., 2014, 2017). However, plagued by the inaccuracy of terminal time prediction, it is difficult for

convex optimization algorithms to solve the problem of free terminal time. According to six flight durations estimated off-line, G-FOLD constructs six trajectories onboard and selects the most fuel-efficient one as the guidance command. The NLP problem is even harder to convexify when the variance in gravitational acceleration based on height and the aerodynamic forces are considered. Therefore, successive convex programming (SCP) was proposed (Lu and Liu, 2013; Liu and Lu, 2014), where the emphasis was placed on the treatment of the nonlinear equations of motion. Two methods, direct linearization (inspired by the linearization of sequential quadratic programming (SQP)) and nonlinearity-kept and linearization (Yang and Liu, 2019), were applied to landing problems with free terminal time or aerodynamic forces (Casoliva, 2013; Szmuk and Açıkmeşe, 2016; Liu, 2019). Mao et al. (2016, 2017, 2018) analyzed the continuous-time convergence and optimality of SCP by combining the trust region and virtual control variables. Zhao and Song (2017) used the terminal time as a control variable, which was combined with the SCP algorithm to obtain fast reentry trajectory planning under multiple constraints. These methods all use convex optimization to solve the relaxed second-order cone constraints of the thrust amplitude. Only when the minimum fuel is taken as the index and the modulus value of the thrust vector is equal to the relaxation variable, can the convexification be equivalent to the original problem. However, the above treatment leads to the bang-bang control, which does not adapt well to the disturbances.

Some constraints, such as the line-of-sight and glide-slope, are coupled with the translational and rotational motion of the spacecraft during landing. Lee and Mesbahi (2015, 2017) proposed an expression of a six-degree-of-freedom (6-DoF) convex optimization problem based on dual quaternions. On this basis, a time-free 6-DoF SCP method was proposed (Szmuk et al., 2017; Szmuk and Açıkmeşe, 2018), and the state trigger constraints (STCs) were introduced into the landing problem combined with the linear compensation problem (Szmuk et al., 2019), which further expanded the application of SCP. For the 6-DOF problem with free terminal time, the SCP algorithm also needs better initial values to converge. At present, the theoretical proof of the optimal convergence is still under research.

Although it is still challenging to prove the convergence of many direct methods for NLP problems, they are effective ways to analyze planetary landing features and have developed in many directions, such as obstacle avoidance (Ma et al., 2016; Zhang B et al., 2016; Wang C and Song, 2018b), vertical takeoff and vertical landing (VTVL) simultaneous optimization (Ma et al., 2018a), and pseudospectral discrete convex optimization (Sagliano and Mooij, 2018; Sagliano, 2018a, 2018b; Wenzel et al., 2018; Malyuta et al., 2019). Simulations have shown that a reasonable initial value can help quickly find the (local) optimal solution of an NLP problem, and the computing speed and convergence effect are not inferior to the counterparts of the convex optimization methods. Hence, how to obtain a good initial guess (Ma et al., 2017, 2019) becomes one of the research directions. Additionally, by analyzing the optimal thrust amplitude profile of planetary landing, Lu (2018) proposed a fast trajectory-generation method by adaptively determining the initial conditions of the PD phase.

When the real-time performance of trajectory planning is guaranteed, the strategies based on receding horizon control (RHC) or model predictive control (MPC) can generate real-time control commands that satisfy the required constraints (García et al., 1989; Mayne et al., 2000; Zeilinger et al., 2014; Pascucci et al., 2015; Lee and Mesbahi, 2017). The control precision of MPC relies on the prediction model, and the optimization time will directly affect the interval of the control cycle. The zero-effort-miss/zero-effort-velocity (ZEM/ZEV) feedback guidance law can also generate an acceleration control command during landing (Furfaro et al., 2011; Ebrahimi et al., 2008), which can guide the lander to track discrete waypoints planned online, avoid obstacles through a multiphase design (Guo et al., 2013; Zhou and Xia, 2014; Zhang Y et al., 2017), and generate a feedback guidance command with terminal attitude constraints by introducing auxiliary control variables (Song et al., 2015; Zhao et al., 2015). However, the ZEM/ZEV method can derive only the guidance command when omitting the aerodynamic forces. Recently, the reinforcement learning based guidance algorithm has also been adopted to enhance the robustness of the ZEM/ZEV algorithms (Furfaro and Linares, 2017; Gaudet et al., 2018; Jiang et al., 2018), but its effectiveness is affected by the training samples, and it is

not adaptable to model uncertainties.

In 2015, SpaceX successfully completed a vertical landing of its rocket booster (Blackmore, 2016), which expanded the application of AGMs. Unlike vacuum or rarefied atmospheric environments, the atmospheric influence on the Earth's surface cannot be neglected. At the same time, the mass of the rocket during landing significantly changes, the larger slenderness ratio demands a stringent attitude constraint, and the terminal landing accuracy is within several meters of the landing site. Today, studies of AGMs in terms of the recovery of rockets have focused mainly on the convex optimization algorithm during the PD phase (Wang JB and Cui, 2018; Wang C and Song, 2018a; Liu, 2019), where the remaining time is relatively short (around 10 s) and the ability to adjust is limited. In summary, the multi-phase simultaneous planning AGM is a key direction of future research (Ma et al., 2018b).

Many space agencies and private companies are also actively engaged in the testing and verification of reusable rockets, such as FROG of the Centre National d'Études Spatiales (CNES) in France (Monchaux et al., 2018), EAGLE of the Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR) in Germany (Dumke et al., 2017; Wenzel, 2017; Sagliano et al., 2019a), RV-X of the Japan Aerospace Exploration Agency (JAXA) (Nonaka, 2018; Sato et al., 2018), and Cooperative Action Leading to Launcher Innovation in Stage Tossback Operations (CALLISTO) (jointly developed by CNES, DLR, and JAXA and planned to fly by 2022 (Dumont et al., 2018; Sagliano et al., 2019b)). One of the main technological G&C branches that are currently under investigation by DLR and JAXA for CALLISTO is convex optimization. Similar research has been carried out by the China Academy of Launch Vehicle Technology (CALT), whose Peacock vehicle validated their online trajectory planning technology at low altitudes and low speeds in 2018. The VTVL vehicle of CALT, composed of four rocket engines, is scheduled for tests under high thrust-to-weight ratio conditions in 2020. The private company LinkSpace validated a state predictive neural network control algorithm on two small vehicles RLV-T3 and RLV-T5 (Chen SZ et al., 2019).

In the following sections, the research status, features, and challenges of AGMs are summarized for lunar landing, Mars landing, and rocket recovery. In the Appendix, all methods for powered planetary landings mentioned in this paper are summarized and classified into five categories.

2 Problem description

2.1 3- and 6-DoF equations of motion

The 3-DoF equation of translational motion describing the powered planetary landing problem is

$$\begin{cases} \dot{\boldsymbol{r}} = \boldsymbol{v}, \\ \dot{\boldsymbol{v}} = \boldsymbol{F}/m - 2\boldsymbol{\omega}_{\mathrm{p}} \times \boldsymbol{v} - \boldsymbol{\omega}_{\mathrm{p}} \times (\boldsymbol{\omega}_{\mathrm{p}} \times \boldsymbol{r}), \\ \boldsymbol{F} = \boldsymbol{D} + \boldsymbol{T} + \boldsymbol{G}, \\ \dot{m} = -\|\boldsymbol{T}\|/(I_{\mathrm{sp}} g_0), \end{cases} \tag{1}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ represent the position and velocity vectors, respectively, and $\boldsymbol{F}$ represents the resultant force on the vehicle, including gravitation $\boldsymbol{G}$, engine thrust $\boldsymbol{T}$, and aerodynamic drag forces $\boldsymbol{D}$. $\boldsymbol{\omega}_{\mathrm{p}}$ is the angular velocity of the rotation of the planet, m is the mass of the vehicle, I_{sp} is the engine-specific impulse, and g_0 is the acceleration of gravity at sea level.

In terms of the attitude-related processes and terminal constraints, the 3-DoF equations of motion can meet only the attitude constraints by assuming that the thrust vector of the engine always coincides with the body axis of the vehicle. The 6-DoF equations of motion can describe the landing process more accurately by modeling the inertia and angular motion. To avoid the singularity caused by 90° of attitude, quaternions are used to describe the attitude of the vehicle. The equation of rotational motion is

$$\begin{cases} \dot{\boldsymbol{\omega}} = \boldsymbol{J}^{-1} \cdot (\boldsymbol{M} - \boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega}), \\ \dot{\boldsymbol{q}} = \dfrac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}) \cdot \boldsymbol{q}, \end{cases} \tag{2}$$

where $\boldsymbol{\omega}$ is the attitude angular velocity, $\boldsymbol{M}$ is the moment, $\boldsymbol{J}$ is the inertia, $\boldsymbol{q}$ is the quaternion vector, and

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}.$$

Eq. (2) ignores the coupling between the angular velocity and inertia along three axes, and the effect of centroid movement on the inertia caused by mass consumption. To simplify the 6-DoF equation of

motion, the dual quaternion method can be used here to describe the translational and rotational motions (Lee and Mesbahi, 2017).

2.2 Characteristic comparison of different scenarios

The atmospheric environment is considered first. The Moon's surface is vacuum and the atmosphere near the surface of the Mars is very thin, so the aerodynamic effects in these two scenarios are usually negligible. The Earth's atmospheric environment is complex and includes wind disturbances. However, atmospheric effects are well understood, and relatively accurate aerodynamic force coefficients and deviation thresholds can be obtained. The aerodynamic force $\boldsymbol{D}$ can be calculated with

$$\boldsymbol{D} = -\frac{1}{2}\rho C_{\mathrm{D}} S_{\mathrm{ref}} \|\boldsymbol{v}\| \boldsymbol{v}, \tag{3}$$

where ρ is the atmospheric density, C_{D} is the aerodynamic force coefficient, and S_{ref} is the reference area.

The landing accuracy of a rocket on the Earth's surface needs to be very high, especially in the scenarios that require landing on a drone ship, whereas the terminal position constraints in the lunar and Mars landing scenarios can be relaxed. The terminal constraints are

$$\begin{cases} \|\boldsymbol{r}(t_f) - \boldsymbol{r}_f\| \le \varepsilon_{\boldsymbol{r}}, \\ \|\boldsymbol{v}(t_f) - \boldsymbol{v}_f\| \le \varepsilon_{\boldsymbol{v}}, \\ \|\phi(t_f) - \pi/2\| \le \varepsilon_{\phi}, \end{cases} \tag{4}$$

where t_f, $\boldsymbol{r}_f$, $\boldsymbol{v}_f$, and ϕ represent the terminal time, target landing position, expected velocity, and pitch angle, respectively. The thresholds ε can be different for each scenario.

In most cases, the Earth's landing site does not include complex terrain, and measurement instruments can be deployed near the site to correct the motion parameters. For lunar and Mars landing, obstacle avoidance is needed using onboard sensors, which introduces the line-of-sight and glide-slope constraints given below:

$$\begin{cases} \cos\theta_{\max} \cdot \|\boldsymbol{T}\| \le T_y, \\ \cos\gamma_{\max} \cdot \|\boldsymbol{r} - \boldsymbol{r}_f\| \le r_y - r_{yf}, \end{cases} \tag{5}$$

where $\theta_{\max}$ and $\gamma_{\max}$ represent the maximum line-of-sight angle and half cone angle of glide-slope, respectively.

According to Eq. (1), the fuel consumption rate is proportional to the thrust amplitude, so the upper and lower limits of the fuel consumption rate are used to restrict the thrust amplitude, as shown in Eq. (6). A high-thrust rocket engine has a limited throttling capacity and fast fuel consumption, while the relatively low thrust engines of planetary landers have deep throttling ability and less fuel consumption.

$$\mathrm{d}m_{\min} \le \dot{m} \le \mathrm{d}m_{\max} \Leftrightarrow T_{\min} \le \|\boldsymbol{T}\| \le T_{\max}. \tag{6}$$

A planetary lander is usually a small vehicle with a slenderness ratio of about 1, and its attitude seldom diverges. However, the rocket has strict attitude constraints. When the roll motion is ignored, the pitch and yaw angular rate and angular constraints are

$$\begin{cases} \phi_{\min} \le \phi \le \phi_{\max}, \\ \psi_{\min} \le \psi \le \psi_{\max}, \\ \omega_{\min} \le \omega_{\phi,\psi} \le \omega_{\max}, \end{cases} \tag{7}$$

where ψ is the yaw angle and $\omega_{\phi,\psi}$ is the pitch and yaw angular rate.

For all landings, the boundary constraints, including the position and velocity equality constraints of the starting point, and the terminal minimum residual mass constraint should be satisfied:

$$\begin{cases} [\boldsymbol{r}, \boldsymbol{v}](t_0) = [\boldsymbol{r}_0, \boldsymbol{v}_0], \\ m(t_f) \ge m_{\min}, \end{cases} \tag{8}$$

where t_0, $\boldsymbol{r}_0$, and $\boldsymbol{v}_0$ represent the initial time, actual position, and velocity at the starting point, respectively.

2.3 Performance index

Less fuel consumption is the most common choice as a performance index. A landing point nearer to the target is acceptable when the fuel quantity is insufficient for the vehicle to reach its original target. In SCP, the virtual control variables and trust region radii are minimized by introducing regularization terms. A quadratic objective function is formed in MPC to track the state and control variables.

The fuel-optimality objective is equivalent to maximizing the terminal residual mass. As shown in Eq. (9), energy-optimality or the shortest flight time may also be used as optimization objectives (J_1, J_2,

J_3, J_4), and J_5 represents the minimum deviation of the landing position if the fuel is insufficient to reach the target point (Blackmore et al., 2010).

$$\begin{cases} J_1 = \int_{t_0}^{t_f} -\dot{m}\,(t)\,\mathrm{d}t, \\ J_2 = -m\,(t_f)\,, \\ J_3 = \int_{t_0}^{t_f} \|\boldsymbol{T}\,(t)\|\,\mathrm{d}t, \\ J_4 = t_f, \\ J_5 = \|\boldsymbol{r}\,(t_f) - \boldsymbol{r}_f\|^2\,. \end{cases} \tag{9}$$

For MPC problems, the minimum tracking deviation is usually used as the objective function:

$$J_{\mathrm{MPC}}=\int_{t_0}^{t_f}\left(\Delta\boldsymbol{X}\,(t)\,\boldsymbol{Q}\Delta\boldsymbol{X}\,(t)+\Delta\boldsymbol{U}\,(t)\,\boldsymbol{R}\Delta\boldsymbol{U}\,(t)\right)\mathrm{d}t, \tag{10}$$

where $\Delta\boldsymbol{X}$ ($\Delta\boldsymbol{U}$) represent the deviations between the state and control variables (target values), and $\boldsymbol{Q}$ and $\boldsymbol{R}$ are weighted matrices.

2.4 General expression

Combined with different performance indices and equations of motion, a general powered planetary landing problem is provided:

$$\begin{aligned} \min\quad & \text{Eq. (9) or (10)} \\ \text{s.t.}\quad & \text{Dynamics: Eq. (1) or Eqs. (1) and (2),} \\ & \text{Constraints: Eqs. (4)–(8).} \end{aligned} \tag{11}$$

The following can be seen from problem (11):

1. Since the aerodynamic force is negligible in lunar landing, an analytical or tracking algorithm can be adopted, and a direct method of online planning can also be used when considering obstacle avoidance or selecting an optimal landing site.

2. For a pinpoint soft landing on Mars, the initial conditions of the powered descent phase cannot be determined in advance due to aerodynamic deceleration, so online planning, such as convex optimization, is needed. However, the aerodynamic force during the PD phase can be ignored, so the planning can be performed only once, and then the tracking method could be used.

3. The rocket vertical landing on Earth is the greatest challenge, and the aerodynamic force cannot be ignored. The nonlinear relationship between the atmospheric density and the altitude in Eq. (3), the limited engine throttling range, and the large mass flow rate all increase the difficulty of online planning. SCP or other direct methods could be alternative solutions, and iterative optimization is needed throughout the process.

Landing on asteroids (Ge et al., 2019) can be regarded as a special case of problem (11).

3 Autonomous guidance methods

The analytical guidance method was the earliest AGM that was studied and applied, where the landing problem was reasonably simplified according to the mission features to obtain guidance laws related to the real-time flight state. In this section, analytical guidance methods, including the Apollo polynomial guidance, optimal guidance based on the maximum principle, and ZEM/ZEV feedback guidance, are introduced. Then, the landing problem is discretized into an NLP problem, and convex optimization and other numerical algorithms are discussed. Finally, a learning-based intelligent algorithm is introduced.

3.1 Analytical guidance methods

With respect to the difficulties in deriving the high-order nonlinear optimal control law by the variational method and the maximum principle, the analytical guidance command can be obtained only by simplifying problem (11), such as neglecting aerodynamic drag, assuming constant mass, uniform gravitational field, and unrestricted thrust change. Three typical methods are introduced here, and other analytical methods can be developed based on them.

3.1.1 Polynomial guidance

The Apollo guidance law (Klumpp, 1974) assumes that acceleration a is a quadratic polynomial of time, where the velocity and position are obtained by integration. By considering the terminal position, and the velocity and acceleration constraints, analytical expressions of the acceleration in line with the current and terminal states are derived, as shown in Eq. (12). The time-to-go is expressed by t_f.

$$\begin{cases} a\,(t) = C_0 + C_1 t + C_2 t^2, \\ v\,(t) = v_0 + C_0 t + \dfrac{1}{2}C_1 t^2 + \dfrac{1}{3}C_2 t^3, \\ r\,(t) = r_0 + v_0 t + \dfrac{1}{2}C_0 t^2 + \dfrac{1}{6}C_1 t^3 + \dfrac{1}{12}C_2 t^4, \\ a\,(t_f) = a_f,\ v\,(t_f) = v_f,\ a\,(t_f) = a_f. \end{cases} \tag{12}$$

From Eq. (12), we can obtain

$$\begin{cases} C_0 = a_f - \dfrac{6}{t_f}(v_f - v_0) + \dfrac{12}{t_f^2}(r_f - r_0 - \dot{r}_0 t_f), \\ C_1 = -\dfrac{6}{t_f} a_f + \dfrac{30}{t_f^2}(v_f - v_0) - \dfrac{48}{t_f^3}(r_f - r_0 - \dot{r}_0 t_f), \\ C_2 = \dfrac{6}{t_f^2} a_f - \dfrac{24}{t_f^3}(v_f - v_0) + \dfrac{36}{t_f^4}(r_f - r_0 - \dot{r}_0 t_f). \end{cases}$$

The guidance method based on higher-order polynomials was developed. The polynomial coefficient vector is regarded as the variable to be optimized. By solving the quadratic optimal control problem with the energy-optimality objective, a polynomial expression related only to t_f is obtained:

$$\begin{aligned} &\begin{cases} J = \int_0^{t_f} \boldsymbol{a}^{\mathrm{T}}(t)\boldsymbol{a}(t)\mathrm{d}t, \\ \boldsymbol{AC} = \boldsymbol{b}, \end{cases} \\ \Rightarrow &\begin{cases} \boldsymbol{a}(t) = \boldsymbol{t}^{\mathrm{T}}\boldsymbol{C}, \\ \boldsymbol{C} = \boldsymbol{S}^{-1}\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{A}\boldsymbol{S}^{-1}\boldsymbol{A}^{\mathrm{T}}\right)^{-1}\boldsymbol{b}, \\ \boldsymbol{S} = \int_0^{t_f} \boldsymbol{t}\boldsymbol{t}^{\mathrm{T}}\mathrm{d}t, \end{cases} \end{aligned} \tag{13}$$

where $\boldsymbol{t} = [\boldsymbol{I}, t\boldsymbol{I}, \ldots, t^N\boldsymbol{I}]^{\mathrm{T}}$, $\boldsymbol{C} = [C_0, C_1, \ldots, C_N]^{\mathrm{T}}$, $\boldsymbol{b} = [a_f, v_f - v_0, r_f - r_0 - v_0 t_f]^{\mathrm{T}}$, and

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{I} & t_f\boldsymbol{I} & \cdots & t_f^N\boldsymbol{I} \\ t_f\boldsymbol{I} & \dfrac{t_f^2\boldsymbol{I}}{2} & \cdots & \dfrac{t_f^{N+1}\boldsymbol{I}}{N+1} \\ \dfrac{t_f^2\boldsymbol{I}}{2} & \dfrac{t_f^3\boldsymbol{I}}{6} & \cdots & \dfrac{t_f^{N+2}\boldsymbol{I}}{(N+1)(N+2)} \end{bmatrix}.$$

The key of the polynomial guidance is to estimate t_f. A linear search algorithm for energy optimality is often used to calculate the optimal time-to-go t_f^*, i.e., estimating the initial value of t_f with the maximum thrust and adopting the Newton method to search for the value of t_f^* that can minimize J in Eq. (13).

China's lunar exploration project has also made successful landings on the moon (Zhang HH et al., 2014a, 2014b). The entire landing of CE-3 includes six stages. In the deceleration stage, the engine-specific impulse and guidance time are estimated based on the linear tangent guidance law (McHenry et al., 1979). In the adjustment stage, the guidance law changes the thrust amplitude and direction linearly to make the lander transit smoothly to the next stage. In the approach stage, a polynomial guidance law that can analytically calculate the guidance time is used. In the hover, obstacle avoidance, and descent stages, the tracking method is adopted for vertical control; after reaching the area above the target point, the lander descends at the lowest possible constant speed.

3.1.2 Guidance methods based on the maximum principle

By considering only longitudinal motion and based on the Pontryagin maximum principle, the lunar landing process with a time-optimality objective comes with full thrust retro-propulsion after free fall (Meditch, 1964); hence, the switching equation was needed. The relationship among the ignition altitude r_y, velocity v_y, and t_f is

$$\begin{cases} r_y = -\dfrac{km_0}{\dot{m}_{\max}} \ln\left(1 - \dfrac{\dot{m}_{\max}}{m_0} t_f\right) - k \cdot t_f - \dfrac{1}{2} g \cdot t_f^2, \\ v_y = k \ln\left(1 - \dfrac{\dot{m}_{\max}}{m_0} t_f\right) + g \cdot t_f, \end{cases} \tag{14}$$

where k represents the velocity of the exhaust gases with respect to the vehicle.

The next step is to approximate the logarithmic terms using quadratic polynomials. If the lander freely falls to a state that meets Eq. (15), the engine ignites and works at full thrust for t_f to reach soft landing:

$$\begin{cases} f(r_y, v_y) = \dfrac{b}{a} r_y + 2a\sqrt{r_y/a} + v_y = 0, \\ t_f = \sqrt{r_y/a}, \end{cases} \tag{15}$$

where $a = \dfrac{1}{2}\dfrac{T_{\max} - gm_0}{m_0}$ and $b = \dfrac{T_{\max}\dot{m}_{\max}}{2m_0^2}$.

3.1.3 ZEM/ZEV feedback guidance law

The ZEM/ZEV feedback guidance law ignores aerodynamic effects. By calculating the time-to-go t_{go}, it solves the guidance command with the energy-optimality objective to meet the position and velocity constraints (Ebrahimi et al., 2008; Furfaro et al., 2011; Guo et al., 2013; Zhou and Xia, 2014; Zhang Y et al., 2017). ZEM and ZEV represent the position and velocity deviations between the lander and the prescribed target at t_{go} if no control force is exerted, respectively:

$$\begin{cases} \mathbf{ZEV}(t) = \boldsymbol{v}_f - \left[\boldsymbol{v}(t) + \int_t^{t_f} \boldsymbol{g}(\tau)\mathrm{d}\tau\right], \\ \mathbf{ZEM}(t) = \boldsymbol{r}_f - \left[\boldsymbol{r}(t) + t_{\mathrm{go}}\boldsymbol{v}(t) + \int_t^{t_f} (t_f - \tau)\boldsymbol{g}(\tau)\mathrm{d}\tau\right], \end{cases} \tag{16}$$

where $t_{\mathrm{go}} = t_f - t$, and $\boldsymbol{g}(\tau)$ represents the acceleration of gravity.

Combined with the objective function and the equation of motion, an analytical expression of the

optimal acceleration command $\boldsymbol{a}_{\mathrm{c}}$ is derived:

$$\begin{aligned}
\min\ & J=\frac{1}{2}\int_t^{t_f}\boldsymbol{a}_{\mathrm{c}}^{\mathrm{T}}\boldsymbol{a}_{\mathrm{c}}\mathrm{d}\tau\\
\text{s.t. }& \mathbf{Z\dot{E}V}=-\boldsymbol{a}_{\mathrm{c}},\\
& \mathbf{Z\dot{E}M}=-\boldsymbol{a}_{\mathrm{c}}t_{\mathrm{go}},\\
& \mathbf{ZEV}(t_f)=\mathbf{ZEM}(t_f)=0.\\
\Rightarrow\ & \begin{cases}\boldsymbol{a}_{\mathrm{c}}=(6/t_{\mathrm{go}}^2)\mathbf{ZEM}-(2/t_{\mathrm{go}})\mathbf{ZEV},\\ \boldsymbol{a}_{\mathrm{c}}=(3/t_{\mathrm{go}}^2)\mathbf{ZEM},\ \ \text{if } \boldsymbol{v}_f \text{ is free},\\ \boldsymbol{a}_{\mathrm{c}}=(1/t_{\mathrm{go}})\mathbf{ZEV},\ \ \text{if } \boldsymbol{r}_f \text{ is free}.\end{cases}
\end{aligned}\tag{17}$$

All three analytical guidance methods include some approximations, which limit their applications. (1) The Apollo and ZEM/ZEV guidance laws assume that the accelerations along the three axes are decoupled, and that the derived acceleration commands are directly related to t_f. It is necessary to make the t_f values of the three axes equal while satisfying the thrust regulation. (2) The longitudinal optimal guidance law derived from the maximum principle guarantees the optimality of bang-bang control using the engine startup equation. However, this equation is derived according to the maximum thrust, which poorly adapts to deviant conditions, especially for Mars landings that requires significant diversions or in the complex aerodynamic environment on Earth. This is also the reason for more than 10-km deviations that occurred during the early Mars landing missions. (3) None of the methods consider process and attitude constraints. The lower limit of the thrust is not considered when calculating the acceleration commands, which means that the engine's thrust amplitude is assumed to be arbitrarily adjustable.

3.2 Numerical optimization algorithms

At present, embedded computers can solve small- and medium-sized NLP problems online using a numerical optimization algorithm (NOA). Even if the landing problem (11) is discretized on each node under the 6-DoF equations of motion, the number of variables to be optimized is smaller than 2000, and the total number of endpoints and process constraints, including the equality constraints of the equation of motion, is still smaller than 3000. This scale is acceptable for online planning. However, the convergence and computational efficiency of numerical optimization methods are still related to the application scenarios, and there is no generalized method.

The basic idea of NOA is to make the problem converge to a (local) optimal solution by iteratively calculating the search direction and step size. Whether the NLP problem is convex or not is the key factor affecting the computing efficiency and convergence. The convergence to global minima is guaranteed theoretically for convex problems, and the high-speed algorithm represented by the primal dual interior point method (PDIPM) can play the role of online programming. However, the convex sets, constraints, and the means of lossless convexification are limited, and convex optimization is not applicable to all problems. Meanwhile, non-convex optimization algorithms can also possibly be used for online planning if suitable initial values are selected.

3.2.1 Online trajectory planning based on convex optimization

Convex optimization refers to a class of optimization problems with convex objective functions, linear equality constraints, and convex inequality constraints. Its advantages include the following: (1) By constructing a homogeneous self-dual embedding model, it does not require users to provide an initial guess of the optimization variables. (2) It has good convergence and polynomial time complexity. (3) It quantifies the approximation between the current value and the optimal solution with the dual gap, and judges the feasibility of the original and dual problems. It includes mainly three categories, linear programming (LP), SOCP, and SDP, all of which require fixed endpoints to the independent variables and monotonic changes throughout the process. Specific forms of convex optimization problems are shown in Table 1, where LP $\subseteq$ SOCP $\subseteq$ SDP.

For the 3-DoF landing problem with constant gravitational acceleration and ignored aerodynamic forces (Açıkmeşe and Ploen, 2007), the glide-slope constraint can be expressed as the second-order cone constraint:

$$\left\|[r_x,r_z]\right\|-\tan(\theta_{\max})\cdot r_y\le 0.\tag{18}$$

The thrust magnitude constraint described by Eq. (6) is non-convex when $T_{\min}>0$, and it can be transformed into a convex constraint by introducing the relaxation variable Γ, as shown in Eq. (19). The

Table 1 Definition of convex optimization

	LP	SOCP	SDP
Description	min $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$, $\boldsymbol{x}\geq\boldsymbol{0}$	min $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}$ s.t. $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$, $\|\boldsymbol{F}_i\boldsymbol{x}-\boldsymbol{d}_i\|\leq\boldsymbol{p}_i^{\mathrm{T}}\boldsymbol{x}+q_i$, $i=1,2,\ldots,l$	min $\mathrm{trace}\,(\boldsymbol{C}\boldsymbol{X})$ s.t. $\mathrm{trace}\,(\boldsymbol{A}_i\boldsymbol{X})=\boldsymbol{b}_i$, $i=1,2,\ldots,m$, $\boldsymbol{X}\succeq\boldsymbol{0}$
Variables	$\boldsymbol{x},\boldsymbol{c}\in\mathbb{R}^n,\boldsymbol{A}\in\mathbb{R}^{n\times n}$, $\boldsymbol{b}\in\mathbb{R}^m,m\leq n$	$\boldsymbol{x},\boldsymbol{c}\in\mathbb{R}^n,\boldsymbol{A}\in\mathbb{R}^{n\times n},\boldsymbol{b}\in\mathbb{R}^m$, $\boldsymbol{F}_i\in\mathbb{R}^{n_i\times n},\boldsymbol{d}_i\in\mathbb{R}^{n_i},\boldsymbol{p}_i\in\mathbb{R}^n,q_i\in\mathbb{R}$	$\boldsymbol{X}\in\boldsymbol{S}^n,\boldsymbol{b}\in\mathbb{R}^m$, $\boldsymbol{C},\boldsymbol{A}_i\in\boldsymbol{S}^n$

LP: linear programming; SOCP: second-order cone programming; SDP: semi-definite programming. $\boldsymbol{S}$: symmetric matrices

optimal solution of the relaxed constraints is equivalent to the original problem with the fuel-optimality objective (Açıkmeşe and Blackmore, 2011):

$$\|\boldsymbol{T}\|\leq\Gamma,\quad T_{\min}\leq\Gamma\leq T_{\max}. \tag{19}$$

For the nonlinear term $\boldsymbol{T}/m$, we define the variable $M=\ln m$ and the new control variable $\sigma=\Gamma/m$. $\boldsymbol{u}=\boldsymbol{T}/m$ while $\|\boldsymbol{u}\|\leq\sigma$. Then the equation of motion can be expressed as

$$\begin{cases}\dot{\boldsymbol{r}}=\boldsymbol{v},\\ \dot{\boldsymbol{v}}=\boldsymbol{u}+\boldsymbol{g}-\left[S\left(\boldsymbol{\omega}_{\mathrm{p}}\right)^2,2S\left(\boldsymbol{\omega}_{\mathrm{p}}\right)\right]\cdot[\boldsymbol{r},\boldsymbol{v}]^{\mathrm{T}},\\ \dot{M}=-\sigma/(I_{\mathrm{sp}}g_0),\\ S\left(\boldsymbol{\omega}_{\mathrm{p}}\right)=\begin{bmatrix}0&-\omega_{x\mathrm{p}}&\omega_{z\mathrm{p}}\\ \omega_{x\mathrm{p}}&0&-\omega_{y\mathrm{p}}\\ -\omega_{z\mathrm{p}}&\omega_{y\mathrm{p}}&0\end{bmatrix}.\end{cases} \tag{20}$$

By taking a Taylor expansion, the thrust amplitude constraint can be converted to a convex constraint:

$$\begin{cases}T_{\min}\mathrm{e}^{-z_0}\left[1-(z-z_0)+0.5\left(z-z_0\right)^2\right]\leq\sigma,\\ \sigma\leq T_{\max}\mathrm{e}^{-z_0}\left[1-(z-z_0)\right],\\ z_0\left(t\right)=\ln\left[m_0-\left(T_{\max}/(I_{\mathrm{sp}}g_0)\right)t\right].\end{cases} \tag{21}$$

Combined with the relaxation variable σ, the line-of-sight constraint can be relaxed to the convex constraint in Eq. (22). Açıkmeşe et al. (2013b) proved that the optimal solution with the J_1 and J_5 objective functions is equivalent to the original problem.

$$\cos(\theta_{\max})\cdot\sigma\leq u_y. \tag{22}$$

To deal with the nonlinear relationships among the aerodynamic forces, gravitational acceleration, and state variables, the SCP algorithm was proposed to guarantee that the solution to the non-convex optimal control problems achieves global convergence with a superlinear convergence rate (Casoliva, 2013; Szmuk and Açıkmeşe, 2016). The SCP algorithm linearizes the nonlinear dynamics, non-convex states, and control constraints near the solution of the previous iteration, and then constructs a convex optimization sub-problem and solves it to obtain the current iteration's solution. Several safe-guarding techniques are incorporated into the algorithm, such as virtual control and buffer technology, to avoid the artificial infeasibility introduced by linearization. The trust region method is used to avoid artificial unboundedness to ensure a solution to each convex optimization sub-problem. Theoretical analyses have shown that if the convergent solution is feasible for the original problem, then it is also the local optimal solution of the current problem (Mao et al., 2016, 2017, 2018).

With the definition of the trust region radius of the state and control variables as $\boldsymbol{\delta}_x,\boldsymbol{\delta}_u>\boldsymbol{0}$, respectively, the trust region constraint can be expressed as

$$\begin{cases}-\boldsymbol{\delta}_x\leq[\boldsymbol{r},\boldsymbol{v},M]-[\boldsymbol{r},\boldsymbol{v},M]^{\kappa}\leq\boldsymbol{\delta}_x,\\ -\boldsymbol{\delta}_u\leq[\boldsymbol{u},\sigma,\boldsymbol{a}_{\mathrm{v}}]-[\boldsymbol{u},\sigma,\boldsymbol{a}_{\mathrm{v}}]^{\kappa}\leq\boldsymbol{\delta}_u,\end{cases} \tag{23}$$

where $\boldsymbol{a}_{\mathrm{v}}$ represents the virtual control variable and κ represents the number of iterations.

By introducing the aerodynamic force acceleration $\boldsymbol{a}_{\mathrm{D}}$ and the virtual control variable $\boldsymbol{a}_{\mathrm{v}}$ into Eq. (20) and linearizing it, we obtain

$$\begin{bmatrix}\dot{\boldsymbol{r}}\\ \dot{\boldsymbol{v}}\\ \dot{M}\end{bmatrix}=\boldsymbol{A}^{\kappa}\begin{bmatrix}\boldsymbol{r}\\ \boldsymbol{v}\\ M\end{bmatrix}+\boldsymbol{B}\begin{bmatrix}\boldsymbol{u}\\ \sigma\\ \boldsymbol{a}_{\mathrm{v}}\end{bmatrix}+\boldsymbol{c}^{\kappa}, \tag{24}$$

where

$$\boldsymbol{B}=\begin{bmatrix}\boldsymbol{0}&\boldsymbol{0}&\boldsymbol{0}\\ \boldsymbol{I}&\boldsymbol{0}&\boldsymbol{I}\\ \boldsymbol{0}&-1/(I_{\mathrm{sp}}g_0)&\boldsymbol{0}\end{bmatrix},$$

$$\boldsymbol{c}^{\kappa}=\begin{bmatrix}\dot{\boldsymbol{r}}^{\kappa}\\ \dot{\boldsymbol{v}}^{\kappa}\\ \dot{M}^{\kappa}\end{bmatrix}-\boldsymbol{A}^{\kappa}\begin{bmatrix}\boldsymbol{r}^{\kappa}\\ \boldsymbol{v}^{\kappa}\\ M^{\kappa}\end{bmatrix},$$

and linearization matrix $\boldsymbol{A}^{\kappa}$ is

$$\begin{cases} \boldsymbol{A}^{\kappa}=\begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{A}_{21}^{\kappa} & \boldsymbol{A}_{22}^{\kappa} & \partial \boldsymbol{a}_{\mathrm{D}}/\partial M^{\kappa} \\ \boldsymbol{0} & \boldsymbol{0} & 0 \end{bmatrix}, \\ \boldsymbol{A}_{21}^{\kappa}=\partial \boldsymbol{g}/\partial \boldsymbol{r}^{\kappa}-S\left(\boldsymbol{\omega}_p\right)^2, \\ \boldsymbol{A}_{22}^{\kappa}=\partial \boldsymbol{a}_{\mathrm{D}}/\partial \boldsymbol{v}^{\kappa}-2S\left(\boldsymbol{\omega}_p\right). \end{cases} \tag{25}$$

For the thrust acceleration amplitude constraint expressed in Eq. (21), the virtual buffer $s_{1,2}$ is

$$\begin{aligned} &T_{\min}\mathrm{e}^{-z_0}\left[1-(z-z_0)+0.5(z-z_0)^2\right]-s_1 \\ &\leq \sigma \leq T_{\max}\mathrm{e}^{-z_0}\left[1-(z-z_0)\right]+s_2, \quad s_{1,2}\in\mathbb{R}_+. \end{aligned} \tag{26}$$

In conjunction with the above-mentioned safeguarding technology, regularization terms need to be added to the above objective functions to minimize the impact of the virtual variables and trust regions on the problem:

$$\begin{aligned} J=J_i&+w_{\mathrm{tr}}\int_{t_0}^{t_f}[\boldsymbol{\delta}_x,\boldsymbol{\delta}_u]^{\mathrm{T}}[\boldsymbol{\delta}_x,\boldsymbol{\delta}_u]\mathrm{d}t \\ &+w_{\mathrm{v}}\int_{t_0}^{t_f}\|\boldsymbol{a}_{\mathrm{v}}(t)\|\mathrm{d}t+w_{\mathrm{s}}\int_{t_0}^{t_f}\|\boldsymbol{s}_{1,2}(t)\|\mathrm{d}t, \end{aligned} \tag{27}$$

where the regularization coefficients w_{tr}, w_{v}, and w_{s} represent the weighted coefficients of the trust region, virtual control variables, and virtual buffer, respectively, $i=1,2,\ldots,5$.

Due to the requirement of a fixed terminal condition for the independent variable in the convex optimization, the above landing problem with the independent time variable is still very sensitive to the estimated accuracy of t_f. Using $t=(t_f-t_0)\tau+t_0$, the time is mapped to the interval $[0,1]$. Taking τ as a new independent variable and t_f as an augmented control variable is an effective way to solve the problem of free terminal time when using the SCP algorithm (Wang C and Song, 2018a). The algorithm is also applicable to the nonlinear equality constraints of the angular velocity and quaternion (Eq. (2)) in the 6-DoF problem (Lee and Mesbahi, 2015, 2017; Szmuk et al., 2017; Szmuk and Açıkmeşe, 2018). Another method is to select other state variables as independent variables, and these variables should be monotonic and have definite boundary values. This method can avoid guessing or specifying the terminal time, which can be calculated with the state equations.

Non-convex process constraints can also be transformed into STCs to form convex constraints with judgment conditions. For example, if the velocity is significant, the bending moment constraint can be expressed as "the angle of attack is less than or equal to the maximum allowable value $\alpha_{\max}$ if the velocity is greater than the threshold v_α":

$$\begin{aligned} &0.5\rho\|\boldsymbol{v}^b\|^2\cdot\alpha\leq q\alpha_{\max} \\ \Rightarrow &-v_x^b(t)\geq\cos(\alpha_{\max})\cdot\left\|\boldsymbol{v}^b(t)\right\|, \text{ if } \left\|\boldsymbol{v}^b(t)\right\|>v_\alpha, \end{aligned} \tag{28}$$

where α denotes the angle of attack and $\boldsymbol{v}^b$ represents the velocity in the vehicle's coordinate system (Szmuk et al., 2019).

With the principle of linear complementarity, the above equation can be transformed into Eq. (29). SCP is used to solve the problem with STC. According to the velocity profile in the previous iteration, h_α is linearized if $g_\alpha<0$; else, we set $h_\alpha=0$.

$$\begin{cases} h_\alpha\left(\boldsymbol{v}^b\right)=-\min\left(g_\alpha\left(\boldsymbol{v}^b\right),0\right)\cdot c_\alpha\left(\boldsymbol{v}^b\right)\leq 0, \\ g_\alpha\left(\boldsymbol{v}^b\right)=v_\alpha-\left\|\boldsymbol{v}^b(t)\right\|, \\ c_\alpha\left(\boldsymbol{v}^b\right)=v_x^b(t)+\cos(\alpha_{\max})\left\|\boldsymbol{v}^b(t)\right\|. \end{cases} \tag{29}$$

The accuracy and sparsity of the discretization method are also key factors that ensure the accuracy and effectiveness of the algorithm. The discretization methods for differential equations of motion include mainly: (1) the trapezoidal method (TM) (Wang C and Song, 2018a), where the estimation of the state variable differential at each discrete point is related to only the state and control variables of two adjacent discrete points; (2) the state transition method (STM) (Açıkmeşe and Ploen, 2007), where the estimation is related to the state and control variables of all previous discrete points; (3) the pseudospectral method (PM) (Sagliano and Mooij, 2018; Sagliano, 2018a, 2018b; Wenzel et al., 2018; Malyuta et al., 2019), where the estimation is related to the control and state variables at all other discrete points. From the approximation accuracy of the discrete problem to the original problem, more information means higher accuracy, so the sequence is PM > STM > TM. From the point of view of sparsity, the left matrix of equality-constrained equations in TM is a banded sparse matrix, while it is the lower triangular matrix in STM and the dense matrix in PM, due to the Lagrange polynomial approximation. So, the alternative sequence when considering sparsity is TM > STM > PM. An intermediate possibility is also given by the hp pseudospectral method

(Sagliano, 2018b), characterized by CPU time similar to that of TM and by limited loss of accuracy with respect to that of PM. In that case, the sparse matrix representing the discretized dynamics is a block-diagonal matrix.

3.2.2 Other numerical optimization-based online planning

1. Homotopy method

The homotopy method is an effective way to solve NLP problems. By setting homotopy parameters and paths, a series of homotopy optimization sub-problems are solved in turn starting from an easy-to-solve problem, where the optimal solution of the current sub-problem is taken as the initial value of the next sub-problem until the solution to the original problem is obtained (Wang C and Song, 2018b). For the optimal trajectory design of VTVL transfer on the lunar surface, Ma et al. (2018a) proposed a simultaneous approach and an adaptive discrete mesh refinement algorithm based on the orthogonal collocation on finite element (OCFE). Although terrain obstacle avoidance constraints in this case complicate the problem, the homotopy backtracking strategy enhances the solvability of the problem. The equation of motion, acting as an equality constraint describing the connections of the state variables at adjacent discrete points, accounts for more than 90% of the total constraints, and the solutions to the sub-problems obtained by the homotopy method satisfy the constraints of the equation of motion, so taking them as the initial values helps solve the NLP problem.

As an example, the discontinuously adjustable thrust constraints can be transformed into inequality constraints $f(T)$:

$$\begin{aligned} &T\in\{0\}\bigcup\left[T_{\min},c_1T_{\max}\right]\bigcup\left[c_2T_{\max},T_{\max}\right]\\ \Leftrightarrow &f(T)=-(T-l_1)\cdot(T-l_2)\cdot(T-l_3)\\ &\qquad\cdot(T-l_4)\cdot(T-l_5)\cdot(T-l_6)\geq 0, \end{aligned} \tag{30}$$

where $l_1=l_2=0$, $l_3=T_{\min}$, $l_4=c_1T_{\max}$, $l_5=c_2T_{\max}$, $l_6=T_{\max}$.

The homotopy sub-problems corresponding to T_1, T_2, and T_3 in Eq. (31) can be solved sequentially, and then the optimal solution satisfying Eq. (30) can be rapidly obtained:

$$\begin{aligned} &T_1=[T_{\min},T_{\max}]\\ \Rightarrow &T_2=[0,c_1T_{\max}]\bigcup\left[c_2T_{\max},T_{\max}\right]\\ \Rightarrow &T_3=\{0\}\bigcup\left[0,c_1T_{\max}\right]\bigcup\left[c_2T_{\max},T_{\max}\right]. \end{aligned} \tag{31}$$

Another typical example is the complex terrain constraints existing with many hills at the landing site:

$$\begin{cases} r\left(t\right)\geq R_{\mathrm{m}}+\left(h_T\left(\theta,\varphi\right)+\Delta h\right), \quad \forall t\in(t_0,t_f),\\ h_T\left(\theta,\varphi\right)=\sum\limits_{i}^{M_T}H_i,\\ H_i{=}h_{Ti}\exp\left(-\dfrac{((\theta-\theta_{0i})R_m)^2}{x_{Ti}}-\dfrac{((\varphi-\varphi_{0i})R_{\mathrm{m}})^2}{y_{Ti}}\right), \end{cases} \tag{32}$$

where R_{m} is the moon radius, M_T is the number of peaks, and h_{Ti} and (θ_{0i},ϕ_{0i}) represent the height and coordinates of each peak, respectively. The homotopy sub-problem without terrain constraints can be solved first, and then the peak constraints are introduced step by step to obtain the optimal solution of the original problem.

2. Sensitivity method

Another method to generate a better initial guess quickly is (parametric) sensitivity analysis. Therein, the Hessian matrix is solved, which represents the optimal condition of Karush-Kuhn-Tucker (KKT) relative to one adjusted variable, and which shows the influence of the change in this variable on the KKT condition (Ma et al., 2017). Ma et al. (2019) proposed an online trajectory optimization algorithm for multi-point powered soft landing on the Mars surface based on NLP sensitivity and an improved K-means clustering algorithm, which improved the accuracy of the fuel consumption estimate by clustering candidate landing sites.

It is important to check the possibility of reaching candidate points to select the optimal one onboard. The position vector $\boldsymbol{r}_f$ of the landing point can be defined as the parameter to be adjusted. According to the principle of the interior point method, the inequality constraints in problem (11) are transformed into logarithmic obstacle terms and added to the objective function. Eq. (33) is then the optimization problem and the corresponding KKT

condition:

$$\begin{aligned}&\min \quad J(\boldsymbol{Z};\boldsymbol{r}_f)-\mu\sum_{i=1}^{n_x}\ln(g_i(\boldsymbol{Z}))\\&\text{s.t.} \quad c(\boldsymbol{Z};\boldsymbol{r}_f)=0.\end{aligned}\tag{33}$$

$$\Rightarrow\begin{cases}\nabla_{\boldsymbol{Z}}L(\boldsymbol{Z},\lambda,\nu;\boldsymbol{r}_f)=\nabla_{\boldsymbol{Z}}J(\boldsymbol{Z};\boldsymbol{r}_f)\\\qquad\qquad+\nabla_{\boldsymbol{Z}}c(\boldsymbol{Z};\boldsymbol{r}_f)\lambda-\nu=0,\\c(\boldsymbol{Z};\boldsymbol{r}_f)=0,\\\boldsymbol{Z}^{(i)}\nu^{(i)}=\mu,\end{cases}$$

where $\boldsymbol{Z}$ represents the vector composed of all state and control variables, μ is the obstacle factor, and λ and ν are Lagrange multipliers.

We first solve for the flight trajectory planning problem corresponding to the approximate geometric center $\boldsymbol{r}_{f0}$ of all landing points. Then, the Hessian matrix corresponding to the KKT condition is expressed as

$$\boldsymbol{H}_{\boldsymbol{s}}(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))=\begin{bmatrix}\boldsymbol{H}_{\boldsymbol{s}11}&\boldsymbol{H}_{\boldsymbol{s}12}&-\boldsymbol{I}\\\boldsymbol{H}_{\boldsymbol{s}21}&\boldsymbol{0}&\boldsymbol{0}\\\boldsymbol{H}_{\boldsymbol{s}31}&\boldsymbol{0}&\boldsymbol{H}_{\boldsymbol{s}33}\end{bmatrix},\tag{34}$$

where $\boldsymbol{s}$ represents all variables to be optimized and

$$\begin{cases}\boldsymbol{H}_{\boldsymbol{s}11}=\nabla_{\boldsymbol{Z}\boldsymbol{Z}}L(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0})),\\\boldsymbol{H}_{\boldsymbol{s}12}=\nabla_{\boldsymbol{Z}}c(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0})),\\\boldsymbol{H}_{\boldsymbol{s}21}=\nabla_{\boldsymbol{Z}}c(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))^{\mathrm{T}},\\\boldsymbol{H}_{\boldsymbol{s}31}=\mathrm{diag}(\nu(\mu;\boldsymbol{r}_{f0})),\\\boldsymbol{H}_{\boldsymbol{s}33}=\mathrm{diag}(\boldsymbol{Z}(\mu;\boldsymbol{r}_{f0})),\\\boldsymbol{s}(\mu;\boldsymbol{r}_{f0})=[\boldsymbol{Z}(\mu;\boldsymbol{r}_{f0}),\lambda(\mu;\boldsymbol{r}_{f0}),\nu(\mu;\boldsymbol{r}_{f0})]^{\mathrm{T}}.\end{cases}$$

Based on the first-order Taylor formula, the initial value of the variables corresponding to other target points can be expressed as

$$\begin{cases}\boldsymbol{s}(0;\boldsymbol{r}_{fi})=\boldsymbol{s}(0;\boldsymbol{r}_{f0})-\boldsymbol{H}_{\boldsymbol{s}}(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))^{-1}\\\qquad\cdot\Big(\boldsymbol{H}_{\boldsymbol{r}_f}(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))\cdot(\boldsymbol{r}_{fi}-\boldsymbol{r}_{f0})\Big),\\\boldsymbol{H}_{\boldsymbol{s}}(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))=\begin{bmatrix}\nabla_{\boldsymbol{Z}\boldsymbol{Z}}L(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))\\\nabla_{\boldsymbol{Z}}c(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))\\0\end{bmatrix},\\\boldsymbol{H}_{\boldsymbol{r}_f}(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))=\begin{bmatrix}\nabla_{\boldsymbol{Z}\boldsymbol{r}_f}L(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))\\\nabla_{\boldsymbol{r}_f}c(\boldsymbol{s}(\mu;\boldsymbol{r}_{f0}))\\0\end{bmatrix}.\end{cases}\tag{35}$$

The optimal solution satisfies all constraints when $\mu=0$. According to $\boldsymbol{r}_{fi}$, the performance index J_i of the corresponding landing point is calculated to quickly find the optimal landing point. Finally, $\boldsymbol{s}(0;\boldsymbol{r}_{fi})$ is used as an initial guess when planning the optimal landing trajectory.

If the underlying optimal control problem is treated as a parameter varying one, a first-order Taylor expansion of the optimal solution under NLP-KKT conditions can be used to derive fast iterative update schemes to approximate the optimal solution for perturbed parameters. In a neighborhood of the nominal optimal solution, this can be used to achieve sub-optimal closed-loop control for nonlinear dynamic systems (Seelbinder, 2017).

3. Numerical predictor-corrector method

For lunar and Mars landing problems with negligible aerodynamic effects, the fuel-optimal thrust is bang-bang if none of the glide-slope, line-of-sight, and thrust regulation rate constraints are considered. The switching between the upper and lower thrust limits ($T_{\max}$ and $T_{\min}$) occurs twice, so the landing problem can be expressed as an optimization problem divided into three stages that have fixed thrust amplitudes (Lu, 2018).

Then the guidance problem is transformed into optimization of the switching time, which greatly reduces the variables' dimensions. By estimating the objective function related to the current switching time, and adopting the univariate minimization method (Brent, 2013) to search for the optimal switching time, which updates only once in each guidance cycle, the online planning is implemented.

3.3 Learning-based method

The learning-based algorithms can achieve the desired control strategies through the neural network, which is trained off-line and is easy to calculate online, to avoid the shortages of convergence and computing efficiency of the numerical methods. However, many training samples are needed which deeply depend on the accuracy of the modeling if the samples are generated by a mathematical model. Two common learning-based methods for powered landing are deep neural networks (Sánchez-Sánchez and Izzo, 2018) and reinforcement learning.

The reinforcement learning process can be summarized as the block diagram in Fig. 1. The lander is regarded as an agent. The state (S) and control (A) variables are consistent with those in the equation of motion, and the output satisfies the constraints

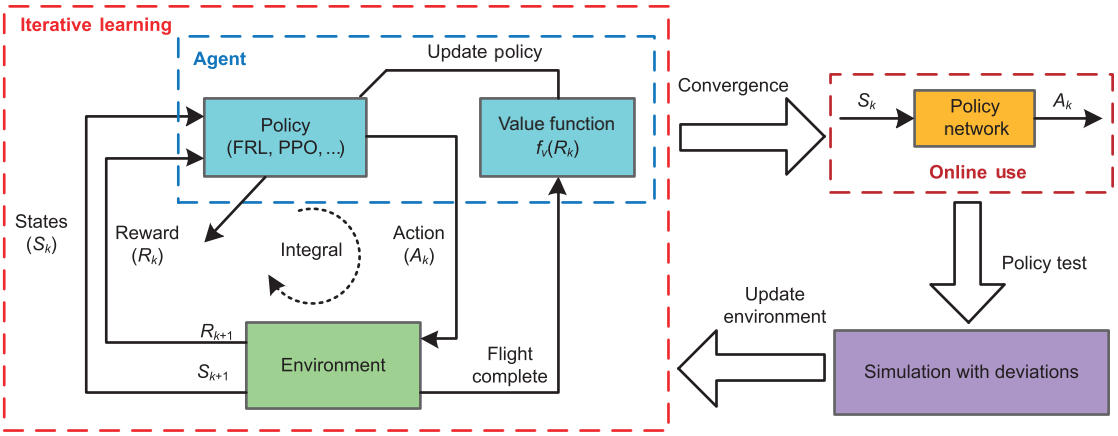

Fig. 1 Reinforcement learning process

in Eqs. (6) and (7). The environment includes the equality constraints of the equations of motion in problem (11) and the process state constraints in Eq. (5). The "reward" and "value" functions are important indices used to update the network, and need to be determined based on the objective functions (Eqs. (9) and (10)) and the terminal constraints. For example, the satisfaction degree of the terminal constraints is taken as the main term of the value function. If the lander satisfies the terminal constraints, it will get a large positive return; otherwise, it will get a large negative return. The return corresponding to the objective function is taken as a small quantity when finding the optimal strategy. Then, combined with the methods of fuzzy reinforcement learning (FRL) (Jouffe, 1998) and proximal policy optimization (PPO) (Schulman et al., 2017), a repeated learning process can be implemented which will converge to the output network under the consideration of possible interferences and uncertainties in different scenarios. During a landing phase, the algorithm outputs control commands in real time according to the current state (Furfaro and Linares, 2017; Gaudet et al., 2018; Jiang et al., 2018).

4 Key issues related to autonomous guidance methods

The convergence and application effects of the above algorithms are related not only to the algorithms themselves, but also to the physical characteristics of the problem. Therefore, this section explores the key problems faced by AGMs from three aspects: physical feasibility, model accuracy, and real-time performance.

4.1 Physical feasibility

Physical feasibility refers to the existence of at least one flight trajectory within the control capability of the lander that can reach the target point from the current state and satisfy all constraints. To ensure a pinpoint landing, the initial state of the PD phase must match the range of the engine thrust regulation and the estimate of t_{go}. The feasible physical region consists of two sets: the reachable set and the controllable set (Benito and Mease, 2010). The controllable set, also known as the access condition, is the focus of the following discussion. It could be searched through trajectory planning algorithms, including direct multiple shooting (Benito and Mease, 2010), lossless converse optimization (Eren et al., 2015), and Hamilton-Jacobi-Isaac traceability analysis (Akametalu et al., 2018).

4.1.1 Access conditions

For a lunar landing with few environmental disturbances, the switching function can be designed according to the thrust amplitude, where there is only one intersection point between the free-falling trajectory of the lander and the switching function. When the lander's state crosses the switching function, it enters the PD phase (Meditch, 1964).

For a Mars landing that requires a large deviation movement, perturbed initial conditions may lead to a PD trajectory that is close to the limit of the lander, or it may not reach the prescribed soft landing location (Lu, 2018). If the planned

fuel-optimal trajectory lies on the boundary of the physically possible flight envelope, a slight deviation will lead to a landing failure. A more suitable access condition is to enter the PD phase as early as possible when the thrust regulation range is satisfied, and the energy-optimal trajectory that meets only the terminal velocity constraint can be optimized to predict the landing point. When the predicted landing point violates the target position, the engine will start to enter the PD phase.

For rocket vertical landings, the aerodynamic drag is greater than or equal to the gravity before entering the PD phase due to the influence of the Earth's dense atmosphere. The velocity along the y-direction should satisfy

$$m_0 g_0 \leq D_y = 0.5\rho C_{\mathrm{D}} S_{\mathrm{ref}} v_{y_0} \left\| \boldsymbol{v}_0 \right\|. \tag{36}$$

First, according to the rocket mass after the first cutoff of the engine, the initial velocity range needs to be predicted when entering the equilibrium state. If only the y-direction motion is considered, the initial height range $[y_{0\,\min}, y_{0\,\max}]$, which can satisfy the terminal velocity and position constraints, can be searched by the Newton method under the conditions of $T_{\min}$ and $T_{\max}$.

Then, traversal of y_0 in $[y_{0\,\min}, y_{0\,\max}]$ needs to be conducted to search the range of the initial position along the x direction. An optimization problem is constructed, as shown in Eq. (37), to ensure that no matter where the lander starts, the deviations of the terminal velocity and position are minimal while maintaining the terminal vertical attitude. A search for x_0 occurs with the Newton method, which allows the performance index in Eq. (37) to gradually approach 0, to obtain the initial position range of the x direction $[x_{0\,\min}, x_{0\,\max}]$ that can satisfy the terminal constraints.

$$\begin{aligned} \min \quad & J = \left(x_f - x_{\exp}\right)^2 + \left(y_f - y_{\exp}\right)^2 \\ & \quad + \left(v_{x_f} - v_{x_{\exp}}\right)^2 + \left(v_{y_f} - v_{y_{\exp}}\right)^2 \\ \text{s.t.} \quad & \text{Dynamics: Eq. (1) or Eqs. (1) and (2),} \\ & \text{Constraints: Eqs. (4)}-\text{(8).} \end{aligned} \tag{37}$$

Subsequently, the method traverses the range of mass $[m_{0\,\min}, m_{0\,\max}]$ and repeats the above process until a set of initial velocities and positions can be obtained off-line as the access conditions of the PD phase. The terminal conditions of the former phase (usually the aerodynamic deceleration phase) are set to be within the access conditions after the vehicle approximately falls down at a constant speed, and as far away as possible from the boundary of the feasible region. For example, we could select the intermediate value as the access condition, so the vehicle can adapt to both positive and negative deviations during landing.

4.1.2 Adaptability of thrust regulation capability to deviations

Assuming that the thrust amplitude is not adjustable, and defining the nominal guidance command as $\phi^*(t)$ and $\psi^*(t)$, the vehicle can swing its body to correct the deviations of the velocity and position. The acceleration of the vehicle is expressed as

$$\begin{cases} \boldsymbol{a}(t) = \dfrac{\|\boldsymbol{T}\|}{m(t)} \left[a_1, a_2, a_3\right]^{\mathrm{T}}, \\ a_1 = \cos\left(\phi^*(t) + \Delta\phi(t)\right) \cos\left(\psi^*(t) + \Delta\psi(t)\right), \\ a_2 = \sin\left(\phi^*(t) + \Delta\phi(t)\right), \\ a_3 = \cos\left(\phi^*(t) + \Delta\phi(t)\right) \sin\left(\psi^*(t) + \Delta\psi(t)\right), \end{cases} \tag{38}$$

where $\Delta\phi$ and $\Delta\psi$ represent the commanded angle corrections. Eq. (38) can be expressed as Eq. (39), when only longitudinal motion is considered:

$$\boldsymbol{a}(t) = \frac{\|\boldsymbol{T}\|}{m(t)} \begin{bmatrix} \cos\phi^*(t) - \Delta\phi(t) \sin\phi^*(t) \\ \sin\phi^*(t) + \Delta\phi(t) \cos\phi^*(t) \end{bmatrix}. \tag{39}$$

Because the pitch angle in the PD phase is near 90°, the effect of guidance command correction on the x-direction acceleration is greater than that on the y-direction acceleration. So, let us take the thrust amplitude as $\|\boldsymbol{T}\|$ and the nominal guidance command corresponding to time t as ϕ^*. When there is a deviation in the y direction, it is necessary to alter the acceleration component in this direction with a corrected commanded angle $\Delta\phi_s$, which causes a deviation of the acceleration in the x direction, i.e., $(T_{xs} - T_{x0})/m$, thus affecting its position or velocity in the x direction. If the thrust amplitude is adjustable in the range of $[T_{\min}, T_{\max}]$, correction of the y-direction deviation can be achieved by simultaneously altering the thrust amplitude and guidance command angle, almost without affecting the x-direction motion. So, the adjustability of thrust regulation is a precondition for a pinpoint landing, where the regulation range determines the adaptability to deviations. $T_{\max}$ and $T_{\min}$ determine the lower

and upper limits of the height in access conditions, respectively. To analyze the relationship between the access conditions and the thrust regulation range (or the thrust-to-weight ratio), the following simulation is considered: the initial mass during the PD phase is 40 000 kg, the maximum thrust of the engine is 1200 kN, the engine-specific impulse is 300 s, and the thrust regulation range is $[\tau, 100\%]$. Supposing that the vehicle has reached the area above the target point by aerodynamic deceleration, i.e., $x = 0$ m, the access conditions of the height-velocity in the longitudinal plane is analyzed by Eq. (37), as shown in Fig. 2. It can be seen that a larger access height leads to a larger feasible velocity region, so it is preferable to enter the PD phase as early as possible; the smaller the thrust regulation range is, the stricter the access conditions are, which means a higher control accuracy is required for the aerodynamic deceleration phase.

The feasible region can also be regarded as the

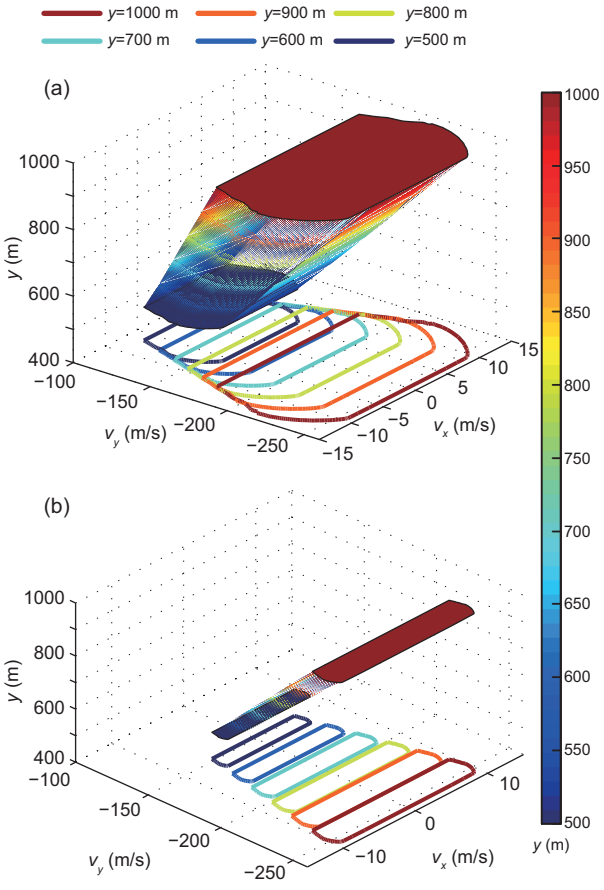

Fig. 2 Access conditions for the powered descent phase: (a) $\tau = 65\%$; (b) $\tau = 90\%$

capacity for adaptation to deviations. If full thrust retro-propulsion is activated and large deviations or disturbances have occurred, a safe landing cannot be assured. Assuming that the fuel remaining for a landing can be determined, the requirement to ensure a successful landing is much more important than saving fuel, so mission planning should aim to expand the feasible physical landing region, and to select access conditions that match the median thrust as the engine start-up condition.

4.2 Model accuracy

Uncertainties in the environment and model should be considered when making decisions during a powered soft-landing process. When the uncertainties are quite prominent, the optimal solution based on the model is usually significantly different from the actual flight process. This is due mainly to the influence of the (limited) measurement accuracy, noise, and other factors on the estimated model parameters, as well as unpredictable dynamical characteristics, resulting in a loss of optimality or even the effectiveness of the solution based on the model. Therefore, we should analyze the compatibility between the optimization solution and model parameters to improve the effectiveness of the optimization results.

Any deviation in the model will be introduced into the optimal control commands generated in each planning cycle. Although the effect can be relieved by receding horizon optimization, it may still shrink the feasible physical region in the landing process due to the limitations of the landing time and thrust regulation range. Taking rocket vertical landing as an example, the aerodynamic drag, mass consumption, and engine thrust amplitude in the model will directly affect the descent acceleration, which therefore affects the accuracy of the rocket's velocity and position. If the command acceleration is less than the actual acceleration, the required thrust magnitude would be gradually reduced by iterative adaptation. However, if the accumulated influence of the model's deviations is large enough, even if the thrust is throttled to a minimum in the follow-up process, it still could not meet the requirements to land, and the velocity will change direction (from descent to ascent) before landing. On the other hand, if the command acceleration is greater than the actual acceleration, that is, the deceleration is insufficient

and the feasible region is gradually reduced until the maximum thrust amplitude is reached, then the vehicle may lose the ability to correct errors and finally land at a higher speed. Therefore, online prediction and model correction are necessary to improve the adaptability of AGM.

An online identification of thrust and aerodynamic drag acceleration deviations can be achieved using a forgetting factor in the recursive least-squares method:

$$\begin{cases}\hat{\boldsymbol{\theta}}(k+1)=\hat{\boldsymbol{\theta}}(k)+\boldsymbol{K}(k+1)\\ \qquad\qquad\cdot\big(\boldsymbol{y}(k+1)-\boldsymbol{x}(k+1)^{\mathrm{T}}\hat{\boldsymbol{\theta}}(k)\big),\\ \boldsymbol{K}(k+1)=\dfrac{\boldsymbol{P}(k)\boldsymbol{x}(k+1)}{\boldsymbol{x}(k+1)^{\mathrm{T}}\boldsymbol{P}(k)\boldsymbol{x}(k+1)+1},\\ \boldsymbol{P}(k+1)=\dfrac{1}{\lambda}\Big(\boldsymbol{P}(k)-\boldsymbol{K}(k+1)\boldsymbol{x}(k+1)^{\mathrm{T}}\boldsymbol{P}(k)\Big).\end{cases}\tag{40}$$

Eq. (41) describes the acceleration along the three axes, including the independent and dependent variables and the parameters to be identified:

$$\begin{cases}\boldsymbol{y}(k)=\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x}(k),\\ \boldsymbol{y}(k)=[a_x,a_y,a_z]^{\mathrm{T}},\\ \boldsymbol{x}(k)=\begin{bmatrix}a_{Tx}&a_{Ty}&a_{Tz}\\ a_{Dx}&a_{Dy}&a_{Dz}\end{bmatrix},\\ \boldsymbol{\theta}=[\tau_{\boldsymbol{T}},\tau_{\boldsymbol{D}}],\end{cases}\tag{41}$$

where $[a_x,a_y,a_z]^{\mathrm{T}}$ represents the real-time measured apparent acceleration, $[a_{Tx},a_{Ty},a_{Tz}]^{\mathrm{T}}$ represents the command acceleration generated by online planning, $[a_{Dx},a_{Dy},a_{Dz}]^{\mathrm{T}}$ represents the estimated aerodynamic drag acceleration based on the model, and $\tau_{\boldsymbol{T}}$ and $\tau_{\boldsymbol{D}}$ represent the thrust and drag coefficient deviations, respectively. $\tau_{\boldsymbol{T}}$ and $\tau_{\boldsymbol{D}}$ are updated in each planning cycle to correct the model.

4.3 Real-time performance

The real-time performance of numerical optimization is a key factor restricting engineering applications. To achieve this performance, it is necessary to ensure the convergence or reliability of the algorithm (such as the convergence of SCP), and then to improve its computational efficiency.

The convergence of the algorithm directly affects the feasibility of AGM. No convergence problem exists in analytic guidance methods, while convex optimization algorithms have mature convergence theory proofs. However, when considering the free terminal time and aerodynamic or process constraints, even for SCP, the solutions are prone to oscillation, and thus convergence is hard to achieve (Yang and Liu, 2019). It is worth pointing out that each convexification method or technique has a great influence on its convergence, and a good initial guess in the non-convex NLP problem will also directly affect its convergence.

However, the traditional convergence criterion, which is reasonable in theoretical applications, is usually very strict and has only two results when terminating an algorithm: success or failure of convergence. By introducing the convergence depth and progress degree according to the convergence state of the current iteration point, the termination of the algorithm could be better controlled (Wang KX et al., 2007; Chen WF et al., 2010).

At present, PDIPM is an important algorithm for solving convex optimization problems for trajectory planning (Domahidi et al., 2012, 2013; Mattingley and Boyd, 2012; Jerez et al., 2017). To further improve PDIPM's computational efficiency, we can customize the solver for small- and medium-sized problems (for fewer than 1000 optimization variables) and generate explicit branches and a loop-free code that can be executed far more quickly, and thus reduce unnecessary computation (Mattingley and Boyd, 2012). However, when the scale of the problem is large, explicit coding significantly increases the amount of code and the size of the executable, which may also negatively affect the computing speed. Moreover, PDIPM automatically generates an initial guess, which also deprives it of an effective means of improving the speed of the numerical algorithm with a warm start. Specifically, in the case of SCP, iteratively updating the initial guess can greatly improve its efficiency.

The alternating direction method of multipliers (ADMM) is a framework for solving large-scale non-convex or convex problems. Although the ADMM algorithm is slower than the gradient descent method and Newton method in terms of convergence speed, the framework is easy to implement in parallel and distributed computing, and can be warm-started (Boyd et al., 2011). Therefore, ADMM can be combined with PDIPM to reduce the number of searches in the Newton direction and step size (Giselsson and Boyd, 2017; Stellato et al., 2018).

Parallel computing or hardware accelerators can

also be used to enhance the real-time performances. For PDIPM, the linear equations that usually need to be solved twice for the affine directions can be carried out in parallel, as can the steps of iteratively updating the optimization and additional variables in ADMM.

5 Vertical landing flight demonstration

National space agencies and companies have designed GNC demonstrators for various vertical landing scenarios. Some are shown in Fig. 3. This section briefly introduces the flight demonstrations of AGMs.

Fig. 3 Flight demonstrator: (a) EAGLE; (b) Peacock

5.1 ADAPT

One of the focuses of the ADAPT project is to validate the G-FOLD algorithm for online Mars landing (Scharf et al., 2014). G-FOLD was translated into the C language and it solves for the online fuel-optimal flight trajectory that satisfies the constraints during the descent of the Xombie vehicle. Large divert was realized and the landing accuracy was controlled at the meter level. The ADAPT project neglected aerodynamic forces, regarded gravitational acceleration as a constant, and validated the real-time performance of the PDIPM-based solver, which could solve 3-DoF SOCP problems online (Açıkmeşe and Ploen, 2007). The customized solver (Dueri et al., 2017) took about 16 ms for each planning on a 1.4-GHz Intel Pentium M processor. Although no optimal flight time was obtained, it evaluated six flight durations, solved six flight trajectories, and selected the optimal solution, where the total planning time was controlled within 100 ms.

5.2 EAGLE

EAGLE's guidance solves a 3-DoF convex optimal control problem discretized by trapezoidal or pseudo-spectral methods into a SOCP that is solved online using the SOCP solver ECOS (Domahidi et al., 2013). To cast the problem in convex form, the minimal flight time is analytically predetermined assuming arcs t_1 and t_2 of constant acceleration (Wenzel, 2017; Wenzel et al., 2018). For each dimension, the minimal flight time is calculated according to Eq. (42) by specifying a maximally desired acceleration:

$$\begin{cases} t_i = t_{1,i} + t_{2,i}, \quad i = x, y, z, \\ t_{1,i} = (-t_{2,i}a_{2,i} - v_{0,i})/a_{1,i}, \\ t_{2,i} = \sqrt{\dfrac{r_{0,i} - r_{f,i} + \dfrac{1}{2}\dfrac{v_{0,i}^2}{a_{1,i}}}{\dfrac{1}{2}a_{2,i}\left(1 - \dfrac{a_{2,i}}{a_{1,i}}\right)}}. \end{cases} \tag{42}$$

The horizontal flight time t_{hor} and the total flight time t_f are then given by

$$\begin{cases} t_{\text{hor}} = \max\left(\sqrt{t_x^2 + t_y^2}, \max\left(t_x, t_y\right)\right), \\ t_f = \max\left(\sqrt{t_{\text{hor}}^2 + t_z^2}, \max\left(t_{\text{hor}}, t_z\right)\right) \cdot k_1 + k_2, \end{cases} \tag{43}$$

where $k_1 \geq 1$, $k_2 \geq 0$ are empirically determined parameters.

To avoid infeasibility due to inaccurate estimation of flight time, the terminal state constraints on position and velocity are formulated in the objective function as quadratic penalty terms. The planning problem is thus always feasible. The terminal conditions are exactly fulfilled if the penalty terms are zero.

For each motion command, EAGLE's guidance algorithm plans the direct motion from the current state to the target state once; additional iterations are possible during the motion to eliminate residuals in the penalty terms. The trajectory is tracked by a 6-DoF controller based on proportional differential control and sliding mode control techniques (Wenzel, 2017; Wenzel et al., 2018). Tethered tests were performed in 2017 and 2018 at the DLR Institute of Space Systems in Bremen to assess and compare the

computational efficiency and accuracy of different variants of the convex guidance algorithm and the performance of the 6-DoF controller.

5.3 Peacock

A low-altitude vertical landing demonstration was carried out with the Peacock vehicle, which was designed by the National Key Laboratory of Aerospace Intelligent Control Technology, a branch of CALT, to simulate limited throttling conditions. With intermediate thrust, the longitudinal velocity-altitude profile for soft landing is calculated off-line, and is fitted with a polynomial as the switching function. Because the descent trajectory intersects only the switching function once (Meditch, 1964), the crossing state is regarded as the ignition condition of the PD phase. By taking fuel-optimality as the objective and considering aerodynamic drag, the terminal time free 3-DoF optimization problem is

$$\begin{aligned} \min \quad & -M_f \\ \text{s.t.} \quad & \dot{\boldsymbol{v}} = t_{\mathrm{go}}\boldsymbol{v}, \\ & \dot{\boldsymbol{v}} = t_{\mathrm{go}}(\boldsymbol{u} + \boldsymbol{D}\mathrm{e}^{-M} + \boldsymbol{g}), \\ & \dot{M} = -t_{\mathrm{go}}\sigma/(I_{\mathrm{sp}}g_0), \\ & \text{Eqs. (19), (21), and (22).} \end{aligned} \tag{44}$$

In combination with MPC strategies, PDIPM-based SCP is implemented onboard iteratively to plan the descent trajectory. t_{go} is estimated first by taking the terminal time of a landing trajectory as the initial guess, and this trajectory is obtained by integration according to the intermediate thrust. To ensure the landing attitude, some variables at the landing time, including angular velocity, speed, and acceleration in the horizontal plane, are reduced as much as possible. The vehicle quickly flies to the region above the target point through a cone constraint with the smallest velocity amplitude in the horizontal plane ($v_x^2 + v_z^2 \leq \eta_v$). The updated objective function is

$$\min \quad -M_f + \sum_{i=0}^{N} i \cdot \eta_v(i), \tag{45}$$

where the weight coefficient i increases gradually as the vehicle approaches the landing point, making the horizontal velocity of the planned trajectory closer to the landing point smaller, so the vehicle can eliminate the deviation in the horizontal plane as soon as possible and move directly above the landing point.

5.4 RLV-T3 and RLV-T5

LinkSpace carried out a low-altitude vertical takeoff-hover-vertical landing test with the RTV-T3 and RLV-T5 vehicles (Chen SZ et al., 2019). Tracking guidance was adopted, and the nonlinear transfer relationship among the engine flow valve's rotation angle u, thrust F, and the mass flow per second was established. According to the equation of longitudinal translational motion, the nonlinear relationship among u, flight speed v, and altitude h was derived:

$$\begin{cases} F(t) = f\left(\int_0^t k_1 u(\tau - \tau_0)\,\mathrm{d}\tau\right), \\ \dot{m}(t) = k_2 \int_0^t k_1 u(\tau - \tau_0)\,\mathrm{d}\tau, \\ v(t) = \int_0^t (F(\tau)/m(\tau) - g)\,\mathrm{d}\tau, \\ h(t) = \int_0^t v(\tau)\,\mathrm{d}\tau, \end{cases} \tag{46}$$

where g represents the value of acceleration of gravity.

A state prediction neutral network control was adopted to replace proportion-integral-differential (PID) to control the flight altitude.

6 Conclusions

In this paper, studies of AGMs for powered planetary landing were reviewed. By comparing the characteristics of three scenarios for landing on the Moon, Mars, and Earth, the constraints and performance index functions for 3- and 6-DoF landing problems were summarized, a general problem description was formed, and three classes of methods, including analytical algorithms, numerical optimization algorithms, and learning-based methods, were discussed in detail. Only SpaceX has validated the vertical landing of rockets and Blue Origin has also successfully performed powered descent landing experiments with their New Shepard rocket, although they were just suborbital flights. However, the efficiency and convergence of a method are deeply affected by engine configurations and the vehicle's characteristics, and soft pinpoint landing is still a challenge for many projects. Although a few demonstrators have been developed and tested, these AGMs need to be verified in real missions. Future research may continue focusing on the appropriate convexification strategy or a smart initial guess for the non-convex problems, and at the same time properly handle the three key issues in

engineering application: physical feasibility, model accuracy, and real-time performance.

Contributors

Zheng-yu SONG guided the AGM research for LM rockets, and organized and revised the manuscript. Cong WANG completed the demonstration algorithm of Peacock, carried out the simulations, and drafted the manuscript. Stephan THEIL, David SEELBINDER, and Marco SAGLIANO provided their first-hand experience on EAGLE and CALLISTO projects, and helped revise the manuscript. Xin-fu LIU revised parts of the manuscript. Zhi-jiang SHAO provided his experience on OCFE and made suggestions to the manuscript.

Compliance with ethics guidelines

Zheng-yu SONG, Cong WANG, Stephan THEIL, David SEELBINDER, Marco SAGLIANO, Xin-fu LIU, and Zhi-jiang SHAO declare that they have no conflict of interest.

References

Açıkmeşe B, Blackmore L, 2011. Lossless convexification of a class of optimal control problems with non-convex control constraints. *Automatica*, 47(2):341-347. https://doi.org/10.1016/j.automatica.2010.10.037

Açıkmeşe B, Ploen SR, 2005. A powered descent guidance algorithm for Mars pinpoint landing. AIAA Guidance, Navigation, and Control Conf and Exhibit, Article 6288. https://doi.org/10.2514/6.2005-6288

Açıkmeşe B, Ploen SR, 2007. Convex programming approach to powered descent guidance for Mars landing. *J Guid Contr Dynam*, 30(5):1353-1366. https://doi.org/10.2514/1.27553

Açıkmeşe B, Aung M, Casoliva J, et al., 2013a. Flight testing of trajectories computed by G-FOLD: fuel optimal large divert guidance algorithm for planetary landing. 23rd AAS/AIAA Spaceflight Mechanics Meeting, Article 386.

Açıkmeşe B, Carson JM, Blackmore L, 2013b. Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem. *IEEE Trans Contr Syst Technol*, 21(6):2104-2113. https://doi.org/10.1109/TCST.2012.2237346

Akametalu AK, Tomlin CJ, Chen M, 2018. Reachability-based forced landing system. *J Guid Contr Dynam*, 41(12):2529-2542. https://doi.org/10.2514/1.G003490

Benito J, Mease KD, 2010. Reachable and controllable sets for planetary entry and landing. *J Guid Contr Dynam*, 33(3):641-654. https://doi.org/10.2514/1.47577

Biegler LT, Zavala VM, 2009. Large-scale nonlinear programming using IPOPT: an integrating framework for enterprise-wide dynamic optimization. *Comput Chem Eng*, 33(3):575-582. https://doi.org/10.1016/j.compchemeng.2008.08.006

Blackmore L, 2016. Autonomous precision landing of space rockets. *Bridge*, 46(4):15-20.

Blackmore L, Açıkmeşe B, Scharf DP, 2010. Minimum-landing-error powered-descent guidance for Mars landing using convex optimization. *J Guid Contr Dynam*, 33(4):1161-1171. https://doi.org/10.2514/1.47202

Blackmore L, Açıkmeşe B, Carson JMIII, 2012. Lossless convexification of control constraints for a class of nonlinear optimal control problems. *Syst Contr Lett*, 61(8):863-870. https://doi.org/10.1016/j.sysconle.2012.04.010

Boggs PT, Tolle JW, 1995. Sequential quadratic programming. *Acta Numer*, 4(4):1-51. https://doi.org/10.1017/S0962492900002518

Bomze IM, Demyanov VF, Fletcher R, et al., 2007. Nonlinear Optimization. Springer, Berlin, Germany. https://doi.org/10.1007/978-3-642-11339-0

Boyd S, Vandenberghe L, 2004. Convex Optimization. Cambridge University Press, New York, USA.

Boyd S, Parikh N, Chu E, et al., 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found Trends Mach Learn*, 3(1):1-122. https://doi.org/10.1561/2200000016

Brent RP, 2013. Algorithms for Minimization without Derivatives. Courier Corporation, New York, USA.

Casoliva J, 2013. Spacecraft Trajectory Generation by Successive Approximation for Powered Descent and Cyclers. PhD Thesis, University of California, Irvine, USA.

Chen SZ, Chu LF, Yang XM, et al., 2019. Application of state prediction neural network control algorithm in small reusable rocket. *Acta Aeron Astron Sin*, 40(3):149-163 (in Chinese).

Chen WF, Shao ZJ, Wang KX, et al., 2010. Convergence depth control for interior point methods. *AIChE J*, 56(12):3146-3161. https://doi.org/10.1002/aic.12225

Domahidi A, Zgraggen AU, Zeilinger MN, et al., 2012. Efficient interior point methods for multistage problems arising in receding horizon control. 51st IEEE Conf on Decision and Control, p.668-674. https://doi.org/10.1109/CDC.2012.6426855

Domahidi A, Chu E, Boyd S, 2013. ECOS: an SOCP solver for embedded systems. European Control Conf, p.3071-3076. https://doi.org/10.23919/ECC.2013.6669541

Dueri D, Jing Z, Açıkmeşe B, 2014. Automated custom code generation for embedded, real-time second order cone programming. 19th Int Federation of Automatic Control World Congress, p.1605-1612. https://doi.org/10.3182/20140824-6-ZA-1003.02736

Dueri D, Açıkmeşe B, Scharf DP, et al., 2017. Customized real-time interior-point methods for onboard powered-descent guidance. *J Guid Contr Dynam*, 40(2):197-212. https://doi.org/10.2514/1.G001480

Dumke M, Sagliano M, Saranrittichai P, et al., 2017. EAGLE - environment for autonomous GNC landing experiments. 10th Int ESA Conf on Guidance, Navigation and Control Systems, p.1-25.

Dumont E, Ecker T, Chavagnac C, et al., 2018. CALLISTO - reusable VTVL launcher first stage demonstrator. Space Propulsion Conf, Article 406.

Ebrahimi B, Bahrami M, Roshanian J, 2008. Optimal sliding-mode guidance with terminal velocity constraint for fixed-interval propulsive maneuvers. *Acta Astron*, 62(10-11):556-562. https://doi.org/10.1016/j.actaastro.2008.02.002

Eren U, Dueri D, Açıkmeşe B, 2015. Constrained reachability and controllability sets for planetary precision landing via convex optimization. *J Guid Contr Dynam*, 38(11):2067-2083. https://doi.org/10.2514/1.G000882

Fahroo F, Ross IM, 2008. Pseudospectral methods for infinite-horizon nonlinear optimal control problems. *J Guid Contr Dynam*, 31(4):927-936. https://doi.org/10.2514/1.33117

Furfaro R, Linares R, 2017. Waypoint-based generalized ZEM/ZEV feedback guidance for planetary landing via a reinforcement learning approach. 3rd IAA Conf on Dynamics and Control of Space Systems, p.401-416.

Furfaro R, Selnick S, Cupples ML, et al., 2011. Nonlinear sliding guidance algorithms for precision lunar landing. 21st AAS/AIAA Space Flight Mechanics Meeting, p.945-964.

García CE, Prett DM, Morari M, 1989. Model predictive control: theory and practice—a survey. *Automatica*, 25(3):335-348. https://doi.org/10.1016/0005-1098(89)90002-2

Gaudet B, Linares R, Furfaro R, 2018. Integrated guidance and control for pinpoint Mars landing using reinforcement learning. *Adv Astron Sci*, 167:3135-3154.

Ge DT, Cui PY, Zhu SY, 2019. Recent development of autonomous GNC technologies for small celestial body descent and landing. *Progr Aerosp Sci*, 110:100551. https://doi.org/10.1016/j.paerosci.2019.06.002

Gill PE, Murray W, Saunders MA, 2005. SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Rev*, 47(1):99-131. https://doi.org/10.1137/S0036144504446096

Giselsson P, Boyd S, 2017. Linear convergence and metric selection for Douglas-Rachford splitting and ADMM. *IEEE Trans Autom Contr*, 62(2):532-544. https://doi.org/10.1109/TAC.2016.2564160

Grant M, Boyd S, Ye Y, 2008. CVX: MATLAB Software for Disciplined Convex Programming. https://cvxr.com/cvx/ [Accessed on Mar. 1, 2020].

Guo YM, Hawkins M, Wie B, 2013. Waypoint-optimized zero-effort-miss/zero-effort-velocity feedback guidance for Mars landing. *J Guid Contr Dynam*, 36(3):799-809. https://doi.org/10.2514/1.58098

Harris MW, Açıkmeş B, 2014. Lossless convexification of nonconvex optimal control problems for state constrained linear systems. *Automatica*, 50(9):2304-2311. https://doi.org/10.1016/j.automatica.2014.06.008

Jerez J, Merkli S, Bennani S, et al., 2017. Forces-RTTO: a tool for on-board real-time autonomous trajectory planning. 10th Int ESA Conf on Guidance, Navigation and Control Systems, p.1-22.

Jiang XQ, Furfaro R, Li S, 2018. Integrated guidance for Mars entry and powered descent using reinforcement learning and Gauss pseudospectral method. 4th IAA Conf on Dynamics and Control of Space Systems, p.761-774.

Jouffe L, 1998. Fuzzy inference system learning by reinforcement methods. *IEEE Trans Syst Man Cybern Part C Appl Rev*, 28(3):338-355. https://doi.org/10.1109/5326.704563

Klumpp AR, 1974. Apollo lunar descent guidance. *Automatica*, 10(2):133-146. https://doi.org/10.1016/0005-1098(74)90019-3

Lee U, Mesbahi M, 2015. Optimal power descent guidance with 6-DoF line of sight constraints via unit dual quaternions. AIAA Guidance, Navigation, and Control Conf, p.1-25.

Lee U, Mesbahi M, 2017. Constrained autonomous precision landing via dual quaternions and model predictive control. *J Guid Contr Dynam*, 40(2):292-308. https://doi.org/10.2514/1.G001879

Liu XF, 2019. Fuel-optimal rocket landing with aerodynamic controls. *J Guid Contr Dynam*, 42(1):65-77. https://doi.org/10.2514/1.G003537

Liu XF, Lu P, 2014. Solving nonconvex optimal control problems by convex optimization. *J Guid Contr Dynam*, 37(3):750-765. https://doi.org/10.2514/1.62110

Lu P, 2017. Introducing computational guidance and control. *J Guid Contr Dynam*, 40(2):193. https://doi.org/10.2514/1.G002745

Lu P, 2018. Propellant-optimal powered descent guidance. *J Guid Contr Dynam*, 41(4):813-826. https://doi.org/10.2514/1.G003243

Lu P, Liu XF, 2013. Autonomous trajectory planning for rendezvous and proximity operations by conic optimization. *J Guid Contr Dynam*, 36(2):375-389. https://doi.org/10.2514/1.58436

Luenberger DG, Ye YY, 1984. Linear and Nonlinear Programming. Springer, New York, USA

Ma L, Shao ZJ, Chen WF, et al., 2016. Trajectory optimization for lunar soft landing with a Hamiltonian-based adaptive mesh refinement strategy. *Adv Eng Softw*, 100:266-276. https://doi.org/10.1016/j.advengsoft.2016.08.002

Ma L, Wang KX, Shao ZJ, et al., 2017. Trajectory optimization for planetary multi-point powered landing. *IFAC-PapersOnLine*, 50(1):8291-8296. https://doi.org/10.1016/j.ifacol.2017.08.1404

Ma L, Wang KX, Xu ZH, et al., 2018a. Trajectory optimization for lunar rover performing vertical takeoff vertical landing maneuvers in the presence of terrain. *Acta Astron*, 146:289-299. https://doi.org/10.1016/j.actaastro.2018.03.013

Ma L, Wang KX, Xu ZH, et al., 2018b. Trajectory optimization for powered descent and landing of reusable rockets with restartable engines. 69th Int Astronautical Congress, Article 44 659.

Ma L, Wang KX, Xu ZH, et al., 2019. Multi-point powered descent guidance based on optimal sensitivity. *Aerosp Sci Technol*, 86:465-477. https://doi.org/10.1016/j.ast.2019.01.028

Malyuta D, Reynolds TP, Szmuk M, et al., 2019. Discretization performance and accuracy analysis for the rocket powered descent guidance problem. AIAA Scitech 2019 Forum, Article 925. https://doi.org/10.2514/6.2019-0925

Mao YQ, Szmuk M, Açıkmeşe B, 2016. Successive convexification of non-convex optimal control problems and its convergence properties. 55th Conf on Decision and Control, p.3636-3641. https://doi.org/10.1109/CDC.2016.7798816

Mao YQ, Dueri D, Szmuk M, et al., 2017. Successive convexification of non-convex optimal control problems with state constraints. *IFAC-PapersOnLine*, 50(1):4063-4069. https://doi.org/10.1016/j.ifacol.2017.08.789

Mao YQ, Szmuk M, Açıkmeşe B, 2018. Successive convexification: a superlinearly convergent algorithm for non-convex optimal control problems. https://arxiv.org/abs/1804.06539v1

Mattingley J, Boyd S, 2012. CVXGEN: a code generator for embedded convex optimization. *Opt Eng*, 13(1):1-27. https://doi.org/10.1007/s11081-011-9176-9

Mayne DQ, Rawlings JB, Rao CV, et al., 2000. Constrained model predictive control: stability and optimality. *Automatica*, 36(6):789-814. https://doi.org/10.1016/S0005-1098(99)00214-9

McHenry RL, de Long AJ, Cockrell BF, et al., 1979. Space shuttle ascent guidance, navigation, and control. *J Astron Sci*, 27:1-38.

Meditch J, 1964. On the problem of optimal thrust programming for a lunar soft landing. *IEEE Trans Autom Contr*, 9(4):477-484. https://doi.org/10.1109/TAC.1964.1105758

Monchaux D, Rmili B, Hassin J, et al., 2018. FROG, a rocket for GNC demonstrations. 69th Int Astronautical Congress, Article 43 308.

Najson F, Mease KD, 2006. Computationally inexpensive guidance algorithm for fuel-efficient terminal descent. *J Guid Contr Dynam*, 29(4):955-964. https://doi.org/10.2514/1.17715

Nonaka S, 2018. Flight demonstration by reusable rocket vehicle RV-X. 28th Workshop on JAXA Astrodynamics and Flight Mechanics, SA6000135029.

Pascucci CA, Bennani S, Bemporad A, 2015. Model predictive control for powered descent guidance and control. European Control Conf, p.1388-1393.

Ploen S, Açıkmeşe B, Wolf A, 2006. A comparison of powered descent guidance laws for Mars pinpoint landing. AIAA/AAS Astrodynamics Specialist Conf and Exhibit, Article 6676. https://doi.org/10.2514/6.2006-6676

Prakash R, Burkhart PD, Chen A, et al., 2008. Mars science laboratory entry, descent, and landing system overview. IEEE Aerospace Conf, p.1-18. https://doi.org/10.1109/AERO.2008.4526283

Sagliano M, 2018a. Pseudospectral convex optimization for powered descent and landing. *J Guid Contr Dynam*, 41(2):320-334. https://doi.org/10.2514/1.G002818

Sagliano M, 2018b. Generalized hp pseudospectral convex programming for powered descent and landing. AIAA Guidance, Navigation, and Control Conf, Article 1870. https://doi.org/10.2514/6.2018-1870

Sagliano M, Mooij E, 2018. Optimal drag-energy entry guidance via pseudospectral convex optimization. AIAA Guidance, Navigation, and Control Conf, Article 1315. https://doi.org/10.2514/6.2018-1315

Sagliano M, Dumke M, Theil S, 2019a. Simulations and flight tests of a new nonlinear controller for the EAGLE lander. *J Spacecr Rock*, 56(1):259-272. https://doi.org/10.2514/1.A34161

Sagliano M, Tsukamoto T, Hernandez J, et al., 2019b. Guidance and control strategy for the CALLISTO flight experiment. 8th EUCASS Conf for Aeronautics and Aerospace Sciences, Article 284. https://doi.org/10.13009/EUCASS2019-284

Sánchez-Sánchez C, Izzo D, 2018. Real-time optimal control via deep neural networks: study on landing problems. *J Guid Contr Dynam*, 41(5):1122-1135. https://doi.org/10.2514/1.G002357

Sato S, Tsukamoto T, Yamamoto T, et al., 2018. The study of navigation, guidance, and control system of reusable vehicle experiment (RV-X). 28th Workshop on JAXA Astrodynamics and Flight Mechanics, SA6000135030.

Scharf DP, Regehr MW, Vaughan GM, et al., 2014. ADAPT demonstrations of onboard large-divert guidance with a VTVL rocket. IEEE Aerospace Conf, p.1-18. https://doi.org/10.1109/AERO.2014.6836462

Scharf DP, Açıkmeşe B, Dueri D, et al., 2017. Implementation and experimental demonstration of onboard powered-descent guidance. *J Guid Contr Dynam*, 40(2):213-229. https://doi.org/10.2514/1.G000399

Schulman J, Wolski F, Dhariwal P, et al., 2017. Proximal policy optimization algorithms. https://arxiv.org/abs/1707.06347

Seelbinder D, 2017. On-board Trajectory Computation for Mars Atmospheric Entry based on Parametric Sensitivity Analysis of Optimal Control Problems. PhD Thesis, Universität Bremen, Bremen, Germany.

Song ZY, Zhao DJ, Lv XG, 2015. Terminal attitude-constrained guidance and control for lunar soft landing. *Adv Astron Sci*, 153:137-147.

Sostaric R, Rea J, 2005. Powered descent guidance methods for the Moon and Mars. AIAA Guidance, Navigation, and Control Conf and Exhibit, Article 6287. https://doi.org/10.2514/6.2005-6287

Stellato B, Banjac G, Goulart P, et al., 2018. OSQP: an operator splitting solver for quadratic programs. https://arxiv.org/abs/1711.08013v2

Szmuk M, Açıkmeşe B, 2016. Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints. AIAA Guidance, Navigation, and Control Conf, Article 378. https://doi.org/10.2514/6.2016-0378

Szmuk M, Açıkmeşe B, 2018. Successive convexification for 6-DoF Mars rocket powered landing with free-final-time. AIAA Guidance, Navigation, and Control Conf, Article 617. https://doi.org/10.2514/6.2018-0617

Szmuk M, Eren U, Açıkmeşe B, 2017. Successive convexification for Mars 6-DoF powered descent landing guidance. AIAA Guidance, Navigation, and Control Conf, Article 1500. https://doi.org/10.2514/6.2017-1500

Szmuk M, Reynolds T, Açıkmeşe B, et al., 2019. Successive convexification for 6-DoF powered descent guidance with compound state-triggered constraints. AIAA Scitech 2019 Forum, Article 926. https://doi.org/10.2514/6.2019-0926

Toh KC, Tutuncu RH, Todd MJ, 2004. On the implementation of SDPT3 (version 3.1) - a MATLAB software package for semidefinite-quadratic-linear programming. IEEE Int Conf on Robotics and Automation, p.290-296. https://doi.org/10.1109/CACSD.2004.1393891

Topcu U, Casoliva J, Mease KD, 2005. Fuel efficient powered descent guidance for Mars landing. AIAA Guidance, Navigation, and Control Conf and Exhibit, Article 6286. https://doi.org/10.2514/6.2005-6286

Topcu U, Casoliva J, Mease KD, 2007. Minimum-fuel powered descent for Mars pinpoint landing. *J Spacecr Rock*, 44(2):324-331. https://doi.org/10.2514/1.25023

Tsiotras P, Mesbahi M, 2017. Toward an algorithmic control theory. *J Guid Contr Dynam*, 40(2):194-196. https://doi.org/10.2514/1.G002754

Wang C, Song ZY, 2018a. Convex model predictive control for rocket vertical landing. 37th Chinese Control Conf, p.9837-9842. https://doi.org/10.23919/ChiCC.2018.8483147

Wang C, Song ZY, 2018b. Rapid trajectory optimization for lunar soft landing with hazard avoidance. *Adv Astron Sci*, 161:885-900.

Wang JB, Cui NG, 2018. A pseudospectral-convex optimization algorithm for rocket landing guidance. AIAA Guidance, Navigation, and Control Conf, Article 1871. https://doi.org/10.2514/6.2018-1871

Wang KX, Shao ZJ, Zhang ZJ, et al., 2007. Convergence depth control for process system optimization. *Ind Eng Chem Res*, 46(23):7729-7738. https://doi.org/10.1021/ie070073s

Wenzel A, 2017. On-board Convex Optimization for Powered Descent Landing of EAGLE. PhD Theis, Lulea University of Technology, Lulea, Sweden.

Wenzel A, Sagliano M, Seelbinder D, 2018. Performance analysis of real-time optimal guidance methods for vertical take-off, vertical landing vehicles. 69th Int Astronautical Congress, Article 44 498.

Wright SJ, 1997. Primal-Dual Interior-Point Methods. Society for Industrial and Applied Mathematics, Philadelphia, USA.

Yang RQ, Liu XF, 2019. Comparison of convex optimization-based approaches to solve nonconvex optimal control problems. AIAA Scitech 2019 Forum, Article 1666. https://doi.org/10.2514/6.2019-1666

Zeilinger MN, Raimondo DM, Domahidi A, et al., 2014. On real-time robust model predictive control. *Automatica*, 50(3):683-694. https://doi.org/10.1016/j.automatica.2013.11.019

Zhang B, Tang S, Pan BF, 2016. Multi-constrained suboptimal powered descent guidance for lunar pinpoint soft landing. *Aerosp Sci Technol*, 48:203-213. https://doi.org/10.1016/j.ast.2015.11.018

Zhang HH, Guan YF, Huang XY, et al., 2014a. Guidance navigation and control for Chang'E-3 powered descent. *Sci Sin Technol*, 44(4):377-384. https://doi.org/10.1360/092014-43

Zhang HH, Liang J, Huang XY, et al., 2014b. Autonomous hazard avoidance control for Chang'E-3 soft landing. *Sci Sin Technol*, 44(6):559-568. https://doi.org/10.1360/092014-51

Zhang Y, Guo YN, Ma GF, et al., 2017. Collision avoidance ZEM/ZEV optimal feedback guidance for powered descent phase of landing on Mars. *Adv Space Res*, 59(6):1514-1525. https://doi.org/10.1016/j.asr.2016.12.040

Zhao DJ, Song ZY, 2017. Reentry trajectory optimization with waypoint and no-fly zone constraints using multiphase convex programming. *Acta Astron*, 137:60-69. https://doi.org/10.1016/j.actaastro.2017.04.013

Zhao DJ, Jiang BY, Lv XG, 2015. Terminal attitude-constrained optimal feedback guidance for pinpoint planetary landing. *Adv Astron Sci*, 153:1689-1696.

Zhou LY, Xia YQ, 2014. Improved ZEM/ZEV feedback guidance for Mars powered descent phase. *Adv Space Res*, 54(11):2446-2455. https://doi.org/10.1016/j.asr.2014.08.011

Appendix: Summary of methods for powered planetary landing

Table A1 Summary of the methods for powered planetary landing

Method	Project or characteristic	Literature	Remark
Analytical method	Tracking guidance or combined with other methods	Song et al. (2015), Zhao et al. (2015), Monchaux et al. (2018), Chen SZ et al. (2019)	T, E, L
	Polynomial guidance	Klumpp (1974), Prakash et al. (2008), Zhang HH et al. (2014a, 2014b)	F, L, M
	ZEM/ZEV feedback guidance	Ebrahimi et al. (2008), Furfaro et al. (2011), Guo et al. (2013), Zhou and Xia (2014), Zhang B et al. (2016) Zhang Y et al. (2017)	M, L
	Pontryagin maximum principle	Meditch (1964), Najson and Mease (2006)	M, L
Indirect method	Accelerated dual gradient projection (QP)	Pascucci et al. (2015)	M
	Pontryagin maximum principle	Topcu et al. (2005)	M
	Predictor-corrector method	Lu (2018)	M
CVX or SCP	G-FOLD software: 100 ms on a 1.4-GHz processor; customized solver; six flight durations estimated off-line; tested by Xombie Rocket	Açıkmeşe and Ploen (2007), Açıkmeşe et al. (2013a), Dueri et al. (2014, 2017), Scharf et al. (2014, 2017)	T, M
	EAGLE of DLR: planned by ECOS with time-to-go estimation; tracked by a 6-DoF controller; tethered test	Dumke et al. (2017), Wenzel (2017), Wenzel et al. (2018), Sagliano et al. (2019a)	T, E
	Semi-definite programming	Açıkmeşe and Ploen (2005)	M
	Second-order cone programming	Açıkmeşe and Ploen (2007)	M
	Minimum-landing-error guidance	Blackmore et al. (2010)	M
	6-DoF optimization	Açıkmeşe et al. (2013b), Lee and Mesbahi (2015, 2017)	M
	CVX with pseudospectral discretization	Sagliano (2018a, 2018b), Sagliano and Mooij (2018), Wenzel et al. (2018), Malyuta et al. (2019)	M
	CVX with state transition discretization	Açıkmeşe and Ploen (2007)	M
	Convex programming with hp pseudospectral method	Sagliano (2018b)	M
	SpaceX reusable booster (CVXGEN solver): planned in a fraction of 1 s; demonstrated by multiple launches	Blackmore (2016)	F, E
	Optimization with aerodynamic drag (SCP)	Szmuk and Açıkmeşe (2016, 2018), Szmuk et al. (2017), Wang JB and Cui (2018), Wang C and Song (2018a), Liu (2019)	M, E
	Contractive sequential convex programming (SCP)	Casoliva (2013)	E, L
	Optimization with state trigger constraints (SCP)	Szmuk et al. (2019)	E
	Peacock (SCP) of CALT: 200 ms on a 1.0-GHz processor; customized code (ECOS); free fly to hundreds of meters	–	T, E
Non-convex direct method	Finite-element collocation approach (IPOPT solver)	Ma et al. (2016, 2018a, 2018b)	E, L
	Fuel-efficient powered descent guidance (SNOPT solver)	Topcu et al. (2007)	M
	Pseudospectral discretization method	Sostaric and Rea (2005)	M, L
	Sensitivity-based trajectory generation (IPOPT solver)	Ma et al. (2017, 2019), Seelbinder (2017)	M
	Adaptive pseudospectral method with hazard avoidance (SNOPT solver)	Wang C and Song (2018b)	L
Learning method	Planetary landing via reinforcement learning	Furfaro and Linares (2017), Gaudet et al. (2018), Jiang et al. (2018)	M
	Optimal landing control via deep neural networks	Sánchez-Sánchez and Izzo (2018)	L

ZEM/ZEV: zero-effort-miss/zero-effort-velocity; QP: quadratic programming; SCP: successive convex programming; G-FOLD: guidance for fuel-optimal large diverts; ECOS: embedded conic solver; DoF: degree-of-freedom; E: landing on Earth; M: landing on Mars; L: landing on the Moon; T: demonstration test; F: actual launch