

# 动力下降制导问题的两阶段序列凸化方法

献给越民义教授 100 华诞

杨沐明<sup>1,2</sup>, 攸国攸<sup>1\*</sup>

1 中国科学院数学与系统科学研究院, 北京 100190;

2 中国科学院大学数学科学学院, 北京 100049

E-mail: ymm@lsec.cc.ac.cn, yougy@amss.ac.cn

收稿日期: 2020-04-02; 接受日期: 2020-05-15; 网络出版日期: 2020-09-08; \* 通信作者

国家自然科学基金 (批准号: 11631013, 11991020, 11991021 和 11971372) 和北京智源人工智能研究院资助项目

**摘要** 可重复使用火箭技术是近五年来航天工程领域的焦点话题. 以三自由度燃料最优控制问题为具体模型, 本文研究火箭回收中的关键问题—动力下降制导问题的求解. 由于此模型中包含非凸动力学约束 (带空气阻力项)、非凸推力大小和方向控制约束等, 直接求解方法难以满足工程上实时性的需求. 为了消除非凸性带来的困难, 本文通过无损凸化、时间离散化和线性化技术将问题转化为一类凸规划问题, 即二阶锥规划问题进行求解. 此外, 为了避免传统的序列凸化策略在求解本文模型时的数值不稳定现象, 本文提出一个新的两阶段序列凸化方法. 数值结果表明所提方法仅需求解少量的凸规划问题, 且拥有比单阶段方法更稳定和高效的数值表现.

**关键词** 火箭 动力下降制导 最优控制 凸化 两阶段

**MSC (2010) 主题分类** 49J15, 90C25

## 1 引言

近年来, 可重复使用火箭技术是航天工程领域十分热门的话题. 传统火箭所采用的一次性助推器在完成助推任务后当即报废, 无法回收利用. 通过对助推器下落过程进行制导控制, 可重复使用火箭技术可实现对火箭助推器的回收再利用. 据估计, 一套成熟的火箭回收方案可使火箭发射节约高达 50% 的成本, 可见这一技术对国家航天实力的增强及太空旅行的普及都具有重要的意义. 2015 年 11 月 24 日, 美国的 Blue Origin 公司成功进行了火箭回收试验, 成为全世界首家实现这一技术的航天公司. 此后, 美国的 SpaceX 公司推出其研制的 Falcon 9 型可重复使用火箭, 仅在 2017 年就成功完成了 13 次陆上和海上回收任务.

作为火箭回收技术的难点之一, 着陆段的精确制导控制是通过求解动力下降制导 (powered descent guidance) 问题来实现的. 其可表示为带有目标函数 (如极小化燃料消耗) 和一系列动力学约束、状态

英文引用格式: Yang M M, You G-Y. A two-stage successive convexification method for the powered descent guidance problem (in Chinese). Sci Sin Math, 2020, 50: 1361–1374, doi: 10.1360/SSM-2020-0105

和控制约束 (如推力大小、方向和变化速率) 的最优控制问题. 通过时间离散化, 这一问题可转化为非线性规划 (nonlinear programming, NLP) 问题进行求解. 由于其中含有非凸的动力学约束和控制约束, 因此求解此 NLP 问题的时间复杂度通常是 NP- 难的. 尽管已有许多针对一般 NLP 问题求解方法的研究, 但因为问题是非凸的, 这些方法往往难以满足可靠性和高效性的工程需求. 特别地, 着陆段制导采用的闭环制导控制方式通常需要在线且实时地求解轨迹优化问题. 极其有限的箭上计算资源和极高的实时性要求给问题的求解带来了极大挑战.

不同于非凸 NLP 问题, 凸规划 (convex programming) 问题因其多项式可解<sup>[1-4]</sup> 的特点而备受航天应用者的青睐. 在过去近 20 年间, 许多学者提出将轨迹优化问题通过序列凸化 (successive convexification) 和无损凸化 (lossless convexification) 转化为一系列凸规划问题进行求解. 2007 年, Acikmese 和 Ploen<sup>[5]</sup> 将火星着陆的三自由度 (three-degree-of-freedom, 3-DoF) 燃料最优动力下降制导问题转化为凸规划问题进行求解, 并首次提出应用于推力大小约束的无损凸化技巧. Blackmore 等<sup>[6]</sup> 在此工作的基础上进一步考虑在没有初始可行轨迹的前提下利用凸规划求解极小化着陆误差的动力下降问题. 此后, 凸规划方法被广泛应用于各类航天领域的问题, 如动力下降制导<sup>[7-12]</sup>、极大横程 (maximum cross range) 轨迹优化<sup>[13]</sup>、再入 (entry) 轨迹优化<sup>[14, 15]</sup>、交会对接 (rendezvous and proximity operations) 轨迹优化<sup>[16]</sup>、无人飞行器 (unmanned aerial vehicle) 路径规划<sup>[17, 18]</sup> 和导弹飞行路径规划<sup>[19]</sup> 等. 根据这类最优控制问题的状态和控制约束特点, 以上研究中的凸规划问题可具体表示为二阶锥规划 (second order cone programming, SOCP) 问题. 在给定解的精度下, 基于内点法 (interior point method, IPM) 的软件如 MOSEK、SeDuMi、SDPT3 和 ECOS 等可以确定 SOCP 问题求解的迭代次数上限. 因此, SOCP 求解算法十分适用于着陆段制导所需的在线实时计算.

本文的研究对象是 3-DoF 燃料最优动力下降制导最优控制问题, 其中包括非凸动力学约束 (带空气阻力项)、非凸推力大小和方向控制约束等. 为了消除非凸性给问题求解带来的困难, 我们通过无损凸化、时间离散化和线性化技术将非凸最优控制问题转化为 SOCP 问题进行求解. 传统的序列凸化策略在求解本文的模型时会出现数值不稳定的现象. 为了应对这一问题并提高算法迭代效率, 我们提出一个新的两阶段序列凸化方法. 数值结果表明所提方法仅需求解少量的凸规划问题, 且两阶段方法拥有比单阶段方法更稳定、高效的数值表现. 本文余下内容安排如下: 第 2 节给出问题背景和最优控制问题模型, 第 3 节介绍模型凸化方法, 求解算法的描述在第 4 节, 数值实验的结果和分析在第 5 节呈现, 第 6 节给出最后的总结.

## 2 问题建模

本节给出垂直起降 (vertical takeoff and landing) 方案下连续时间的燃料最优动力下降制导模型. 我们约定: 标量用斜体表示, 如质量  $m$  等; 矢量用粗斜体表示, 如位置  $\mathbf{r}$ 、速度  $\mathbf{v}$  和推力  $\mathbf{T}$  等; 字母记号上方的点表示物理量对时间的导数, 如  $\dot{\mathbf{r}} = d\mathbf{r}/dt$ ; 范数符号  $\|\cdot\|$  表示 Euclid 范数. 在本文之后的描述中, 我们用“火箭”一词表示需要回收的助推器.

本文考虑的火箭运动模型为 3-DoF 模型, 即将火箭主体视为一个质点且在除推力方向约束外仅考虑火箭的平行移动. 3 维坐标的 3 个维度正方向依次为北 (north, N)、上 (up, U)、东 (east, E). 令  $t \in \mathbb{R}_+$  表示飞行时刻,  $t_f$  表示终端时刻. 在燃料最优的最优控制模型中, 我们希望通过确定各个时刻的推力指令  $\mathbf{T}$  和飞行总时长 (也即终端时刻)  $t_f$  使得火箭着陆时的剩余燃料质量极大化, 即极大化火箭终端质量  $m(t_f)$ . 火箭的飞行需要满足若干动力学约束. 结合 Newton 第二定律可以得到如下的动

力学方程:

$$\dot{m}(t) = -\alpha \|\mathbf{T}(t)\|, \quad (2.1)$$

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t), \quad (2.2)$$

$$\dot{\mathbf{v}}(t) = \frac{1}{m(t)} [\mathbf{T}(t) + \mathbf{D}(t)] + \mathbf{g}, \quad (2.3)$$

其中  $m \in \mathbb{R}_+$ 、 $\mathbf{r} \in \mathbb{R}^3$ 、 $\mathbf{v} \in \mathbb{R}^3$  和  $\mathbf{T} \in \mathbb{R}^3$  分别表示火箭的质量、位置、速度和所受推力. 在飞行过程中, 火箭的质量随着燃料的消耗而逐渐减小. 因此,  $\dot{m}$  反映了燃料消耗速度, 它与发动机输出推力大小有关. 常参数  $\alpha$  和  $\mathbf{g}$  分别为发动机比冲的倒数和重力加速度.  $\mathbf{D}(t) \in \mathbb{R}^3$  表示  $t$  时刻火箭所受气动阻力, 其具体计算公式为

$$\mathbf{D}(t) = -\frac{1}{2} \rho S_D C_D \|\mathbf{v}(t)\| \mathbf{v}(t),$$

其中  $\rho$  为大气密度,  $S_D$  为火箭有效横截面积,  $C_D$  表示气动力轴向系数.

此外, 火箭的动力下降制导还需满足一定的状态和控制约束. 首先, 终端时刻的燃料剩余量必须是非负的, 即

$$m(t_f) \geq m_{\text{dry}}, \quad (2.4)$$

其中  $m_{\text{dry}}$  表示火箭的干质量. 其次, 火箭在飞行过程中既不能与地面相撞, 也不能在除着陆点外的其他位置与地面距离过近以防被地面凸起物干涉飞行. 因此, 将火箭的飞行区域限制在一个锥形区域内 (见图 1), 即

$$\|\mathbf{r}(t)\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}(t), \quad (2.5)$$

其中  $\mathbf{e}_U = [0 \ 1 \ 0]^\top$  表示  $U$  方向的单位向量. 最后, 发动机推力的大小、方向和变化速率也需要一定的限制: 发动机不仅有推力上限, 也有推力下限 (即不能在飞行途中熄火); 推力方向需与箭体呈小角度

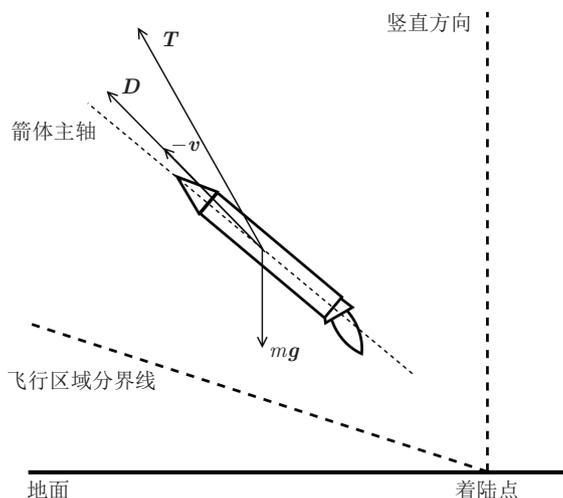


图 1 (网络版彩图) 火箭受力及飞行场景示意图

锐角; 推力大小在单位时间内的变化量是有限的. 以上限制可分别表示为如下的约束:

$$T_{\min} \leq \|\mathbf{T}(t)\| \leq T_{\max}, \quad (2.6)$$

$$\frac{(-\mathbf{v}(t))^{\top} \mathbf{T}(t)}{\|-\mathbf{v}(t)\| \|\mathbf{T}(t)\|} \geq \cos \theta_2, \quad (2.7)$$

$$\dot{T}_{\min} \leq \frac{d\|\mathbf{T}(t)\|}{dt} \leq \dot{T}_{\max}. \quad (2.8)$$

注意到在 3-DoF 模型中, 由于将箭体抽象为质点, 我们无法精确计算攻角. 事实上, 着陆段开始前的调姿操作会将箭体主轴与速度方向调节成接近平行 (见图 1). 因此, 在约束 (2.7) 中, 我们把箭体方向近似为速度反方向  $-\mathbf{v}$ .

除以上约束外, 模型中还必须给定部分状态和控制变量的边界条件, 即初始质量、位置、速度、推力方向及终端位置、速度和推力方向,

$$m(0) = m_{\text{init}}, \quad \mathbf{r}(0) = \mathbf{r}_{\text{init}}, \quad \mathbf{v}(0) = \mathbf{v}_{\text{init}}, \quad \mathbf{T}(0) = \|\mathbf{T}(0)\| \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \quad (2.9)$$

$$\mathbf{r}(t_f) = \mathbf{0}, \quad \mathbf{v}(t_f) = \mathbf{0}, \quad \mathbf{T}(t_f) = \|\mathbf{T}(t_f)\| \mathbf{e}_U. \quad (2.10)$$

在约束 (2.9) 中, 初始的推力方向选取为速度反方向. 终端条件 (2.10) 表示火箭不仅要降落在指定位置, 而且落地速度和横向加速度必须为 0.

令  $\mathbf{X} \in \mathbb{R}^7$  为包含所有状态变量的矢量, 即

$$\mathbf{X}(t) = [m(t) \ \mathbf{r}(t) \ \mathbf{v}(t)].$$

综上所述, 燃料最优动力下降制导的最优控制模型可归结为

$$\begin{aligned} \text{(ROC)} \quad & \min_{\mathbf{X}, \mathbf{T}, t_f} -m(t_f) \\ & \text{s.t. (2.1)–(2.10)}. \end{aligned}$$

注意到 (ROC) 问题是连续时间的最优控制问题, 可通过时间离散化的方式将此问题转化为 NLP 问题进行求解. 此外, (ROC) 问题还包含非凸约束, 如约束 (2.1)、(2.3)、(2.6) 和 (2.8) 等. 下一节通过无损凸化、时间离散化和线性化的方式将 (ROC) 问题转化为凸 SOCP 问题.

### 3 模型转化

#### 3.1 无损凸化

利用参考文献 [5] 等的无损凸化方法, 我们引入新的控制变量  $\Gamma$  表示推力大小的上界, 即

$$\|\mathbf{T}(t)\| \leq \Gamma(t). \quad (3.1)$$

同时将约束 (2.1) 和 (2.6)–(2.10) 中的  $\|\mathbf{T}\|$  替换为  $\Gamma$ . 文献 [5] 证明了在最优解处这种凸化方式是无损的, 即最终得到的解使得 (3.1) 取等号. 虽然目前还不能证明包含气动阻力动力学约束的模型同样可以保持无损性, 但从第 5 节展示的数值结果来看, 这种凸化方式在我们的模型中也是无损的.

### 3.2 时间离散化

将 0 时刻到  $t_f$  时刻的飞行时间均匀分成  $N$  等份, 每份时间长度记为  $\Delta t$ , 同时以下标  $k$  表示第  $k$  个时间点, 即

$$t_f = N\Delta t, \quad \mathbf{X}_k = \mathbf{X}\left(\frac{kt_f}{N}\right), \quad k = 0, 1, \dots, N.$$

当  $N$  确定时, 由于  $t_f$  是变量, 因此  $\Delta t$  也是变量. 微分方程约束 (2.1)–(2.3) 和 (2.8) 可通过差分方法转化为如下有限个线性或非线性约束:

$$\begin{aligned} \mathbf{X}_{k+1} &= \mathbf{X}_k + \mathbf{F}_k(\mathbf{Y}_k), \quad k = 0, 1, \dots, N-1, \\ \dot{T}_{\min}\Delta t &\leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max}\Delta t, \quad k = 0, 1, \dots, N-1, \end{aligned} \quad (3.2)$$

其中矢量值函数  $\mathbf{F}_k(\cdot): \mathbb{R}^{23} \rightarrow \mathbb{R}^7$  和矢量  $\mathbf{Y}_k \in \mathbb{R}^{23}$  分别定义为

$$\mathbf{F}_k(\mathbf{Y}_k) = \frac{\Delta t}{2} \begin{pmatrix} -\alpha(\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[ \frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}} \right] + 2\mathbf{g} \end{pmatrix},$$

$$\mathbf{Y}_k = [\Delta t \ \mathbf{X}_k \ \mathbf{U}_k \ \mathbf{X}_{k+1} \ \mathbf{U}_{k+1}],$$

$$\mathbf{U}_k = [\mathbf{T}_k \ \Gamma_k].$$

### 3.3 线性化

最后, 将非凸动力学约束 (3.2) 和推力方向约束 (2.7) 通过线性化的方式进行凸化. 令

$$G_k(\mathbf{Y}_k) = \mathbf{v}_k^\top \mathbf{T}_k + \cos \theta_2 \|\mathbf{v}_k\| \Gamma_k,$$

可以得到如下线性约束:

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1,$$

$$G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1,$$

其中  $\bar{\mathbf{Y}}_k$  是一组已知轨迹,  $\mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k) \in \mathbb{R}^{7 \times 23}$  和  $\mathcal{J}G_k(\bar{\mathbf{Y}}_k) \in \mathbb{R}^{1 \times 23}$  分别是  $\mathbf{F}_k$  和  $G_k$  在  $\bar{\mathbf{Y}}_k$  处的 Jacobi 矩阵. 在序列凸化方法中,  $\bar{\mathbf{Y}}_k$  可选取为上一次迭代的轨迹, 因此需要在算法开始时给定一条初始轨迹. 合理地选取初始轨迹将有助于减少迭代次数. 我们将在第 4.1 小节给出针对此模型的构造初始解方法.

### 3.4 SOCP 模型

综上, 我们给出 (ROC) 问题经离散化和凸化后的凸规划模型:

$$\begin{aligned} (\text{RSOCP}(\bar{\mathbf{Y}})) \quad & \min_{\mathbf{Y}} \quad -m_N \\ \text{s.t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1, \end{aligned} \quad (3.3)$$

$$T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \quad (3.4)$$

$$\|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \quad (3.5)$$

$$G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1, \quad (3.6)$$

$$\dot{T}_{\min} \Delta t \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta t, \quad k = 0, 1, \dots, N-1, \quad (3.7)$$

$$m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \quad (3.8)$$

$$m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U. \quad (3.9)$$

上述问题仅含线性约束和二阶锥约束, 因此, 是凸 SOCP 问题. 当 (RSOCP) 问题的解  $\mathbf{Y}$  满足以下条件时:

$$\begin{aligned} \mathbf{Y}_k &= \bar{\mathbf{Y}}_k, \quad k = 0, 1, \dots, N-1, \\ \|\mathbf{T}_k\| &= \Gamma_k, \quad k = 1, 2, \dots, N-1, \end{aligned}$$

其可视为 (ROC) 问题的一个离散解.

## 4 模型求解

### 4.1 初始化

我们采用类似文献 [9] 中的方式计算初始轨迹  $\mathbf{Y}^{(0)}$ . 给定终端时刻猜测值  $\tau_f$ , 可得变量  $\Delta t$  的猜测值  $\Delta \tau = \tau_f / N$ . 假定终端质量为干质量  $m_{\text{dry}}$ . 利用质量  $m$  和速度  $\mathbf{v}$  的初始值和终端值, 通过线性插值的方式得到全部  $N+1$  个时刻的质量和速度轨迹, 分别记为  $\mu$  和  $\mathbf{w}$ , 即

$$\begin{aligned} \mu_k &= m_{\text{init}} - \frac{k}{N}(m_{\text{init}} - m_{\text{dry}}), \quad k = 0, 1, \dots, N, \\ \mathbf{w}_k &= \mathbf{v}_{\text{init}} - \frac{k}{N}\mathbf{v}_{\text{init}}, \quad k = 0, 1, \dots, N. \end{aligned}$$

通过将变量  $\Delta t$  近似为  $\Delta \tau$ , 将部分变量  $m$  和  $\mathbf{v}$  近似为  $\mu$  和  $\mathbf{w}$ , 可得到如下的初始化问题:

$$\begin{aligned} (\text{RSOCP-INIT}) \quad & \min_{\mathbf{Z}} \quad -m_N + \beta \|\mathbf{a}^R\| \\ & \text{s.t.} \quad \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{\text{init}}(\mathbf{Z}_k), \quad k = 0, 1, \dots, N-1, \\ & \quad T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \\ & \quad \|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\ & \quad \mathbf{w}_k^\top \mathbf{T}_k + \cos \theta_2 \|\mathbf{w}_k\| \Gamma_k \leq 0, \quad k = 1, 2, \dots, N-1, \\ & \quad \dot{T}_{\min} \Delta \tau \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta \tau, \quad k = 0, 1, \dots, N-1, \\ & \quad m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \\ & \quad m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U, \end{aligned}$$

其中  $\mathbf{a}^R$ 、 $\mathbf{Z}$ 、 $\mathbf{Z}_k$  和  $\mathbf{F}_k^{\text{init}}$  定义为

$$\begin{aligned} \mathbf{a}^R &= [(\mathbf{a}_0^R)^\top (\mathbf{a}_1^R)^\top \cdots (\mathbf{a}_N^R)^\top]^\top, \\ \mathbf{Z} &= [\mathbf{Y}^\top (\mathbf{a}^R)^\top]^\top, \quad \mathbf{Z}_k = [\mathbf{Y}_k^\top (\mathbf{a}^R)_k^\top (\mathbf{a}^R)_{k+1}^\top]^\top, \end{aligned}$$

$$\mathbf{F}_k^{\text{init}}(\mathbf{Z}_k) = \frac{\Delta\tau}{2} \begin{pmatrix} -\alpha(\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[ \frac{\mathbf{T}_k + \tilde{\mathbf{D}}_k}{\mu_k} + \frac{\mathbf{T}_{k+1} + \tilde{\mathbf{D}}_{k+1}}{\mu_{k+1}} \right] + \mathbf{a}_k^R + \mathbf{a}_{k+1}^R + 2\mathbf{g} \end{pmatrix},$$

$$\tilde{\mathbf{D}}_k = -\frac{1}{2}\rho S_D C_D \|\mathbf{w}_k\| \mathbf{v}_k.$$

虽然 (RSOCP-INIT) 问题的目标函数是非线性函数, 但可通过引入新的变量和二阶锥约束进而将问题转化成 SOCP 问题进行求解. 利用参考文献 [9] 的方式, 我们在加速度中引入松弛变量  $\mathbf{a}^R$  作为辅助加速度来避免问题不可行. 注意到当问题给定时, 初始轨迹仅依赖于终端时刻猜测值  $\tau_f$ . 事实上, 我们并不要求  $\tau_f$  与最优轨迹终端时刻很接近, 即算法的收敛性对估计值  $\tau_f$  的准确度依赖程度较低. 不同  $\tau_f$  下的算法迭代次数和最终解的比较将在第 5 节给出.

## 4.2 序列凸化

利用前一小节得到的初始解, 我们可以通过序列凸化的方式迭代地求解 (RSOCP) 问题, 直至相邻两次的轨迹比较接近时终止算法. 简要的算法步骤描述如下:

- (1) 通过初始化得到可行解  $\mathbf{Y}^{(0)}$ ,  $i := 0$ .
- (2) 求解 (RSOCP( $\mathbf{Y}^{(i)}$ )) 问题, 得到解  $\mathbf{Y}^{(i+1)}$ .
- (3) 若  $\|\mathbf{Y}^{(i+1)} - \mathbf{Y}^{(i)}\| < \epsilon$ , 终止; 否则令  $i := i + 1$ , 继续执行步 (2).

在求解 (RSOCP) 问题时, 我们仍采用上一小节给出的引入辅助加速度  $\mathbf{a}^R$  的方式来避免问题不可行. 在实际应用此序列凸化算法进行计算时, 我们发现算法会出现数值不稳定或迭代效率较低的问题, 其原因如下.

- 推力与速度反方向夹角约束给问题求解带来困难: 将约束 (3.6) 从 (RSOCP) 问题中去掉可改善数值不稳定的问题;
  - 变量  $\Delta t$  相较其他变量与真解更接近 (见图 2): 仅需少量迭代就已经满足  $\Delta t$  的最优性条件.
- 参考以上两点, 我们给出如下一小节所述的两阶段序列凸化方法.

## 4.3 两阶段法

根据序列凸化算法存在的问题, 我们设计如下的两阶段方法.

**第一阶段:** 去掉约束 (3.6), 仅在其余约束的基础上进行优化, 直至  $\Delta t$  变化程度较小时, 即条件

$$|\Delta t^{(i)} - \Delta t^{(i-1)}| < \eta$$

满足时切换至第二阶段;

**第二阶段:** 将变量  $\Delta t$  的取值固定为  $\Delta\tau := \Delta t^{(i)}$  并加入约束 (3.6), 继续优化直至满足序列凸化终止条件.

两个阶段的优化问题 (RSOCP-S1) 和 (RSOCP-S2) 分别定义如下:

$$\begin{aligned} \text{(RSOCP-S1}(\bar{\mathbf{Z}})) \quad & \min_{\mathbf{Z}} -m_N + \beta \|\mathbf{a}^R\| \\ \text{s.t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{s1}(\bar{\mathbf{Z}}_k) + \mathcal{J}\mathbf{F}_k^{s1}(\bar{\mathbf{Z}}_k)(\mathbf{Z}_k - \bar{\mathbf{Z}}_k), \quad k = 0, 1, \dots, N-1, \\ & T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \end{aligned}$$

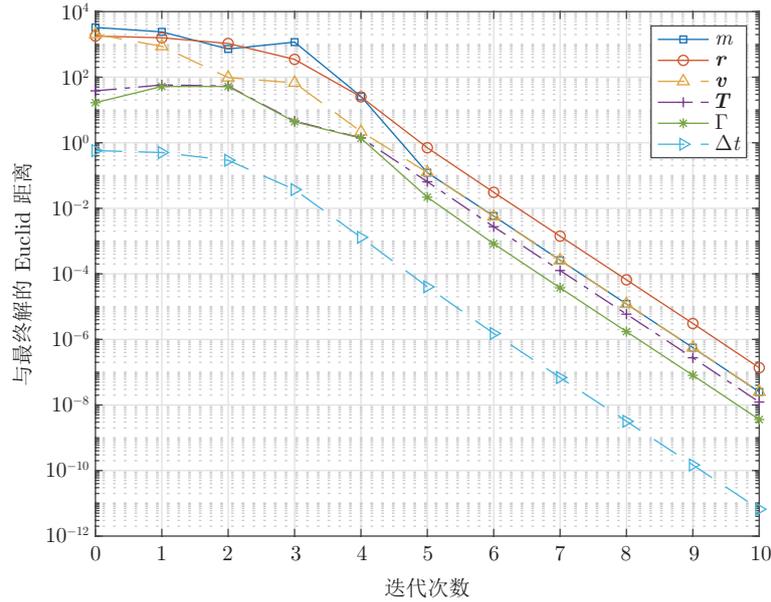


图 2 (网络版彩图) 各变量与最终解的距离

$$\begin{aligned} \|\mathbf{r}_k\| \cos \theta_1 &\leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\ \dot{T}_{\min} \Delta t &\leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta t, \quad k = 0, 1, \dots, N-1, \\ m_0 &= m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \\ m_N &\geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U \end{aligned}$$

和

$$\begin{aligned} (\text{RSOCP-S2}(\bar{\mathbf{Y}})) \quad &\min_{\mathbf{Y}} -m_N \\ \text{s.t.} \quad &\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{\text{s2}}(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k^{\text{s2}}(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1, \\ &T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \\ &\|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\ &G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1, \\ &\dot{T}_{\min} \Delta \tau \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta \tau, \quad k = 0, 1, \dots, N-1, \\ &m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \\ &m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U, \end{aligned}$$

其中  $\mathbf{F}_k^{\text{s1}}$  和  $\mathbf{F}_k^{\text{s2}}$  的定义为

$$\mathbf{F}_k^{\text{s1}}(\mathbf{Z}_k) = \frac{\Delta t}{2} \begin{pmatrix} -\alpha(\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ [\frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}}] + \mathbf{a}_k^R + \mathbf{a}_{k+1}^R + 2\mathbf{g} \end{pmatrix},$$

$$F_k^{s2}(\mathbf{Y}_k) = \frac{\Delta\tau}{2} \begin{pmatrix} -\alpha(\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ [\frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}}] + 2\mathbf{g} \end{pmatrix}.$$

## 5 数值实验

本节将以一个测试算例展示两阶段序列凸化方法求解 (ROC) 问题的数值表现. 我们在 MATLAB 平台实现第 4.2 和 4.3 小节基于序列凸化方法的单阶段算法和两阶段算法, 调用建模工具 `cvx`<sup>[20, 21]</sup> 和求解器 `ECOS`<sup>[22]</sup> 实现 SOCP 问题的建立和求解. 测试环境为一台装有酷睿 i7-7567U 处理器和 16 GB 内存的笔记本电脑. 测试数据 (见表 1) 是在文献 [9] 中测试数据的基础上根据本文研究模型 (如推力方向约束) 修改得到的.

我们以表 1 中的数据建立 (ROC) 问题, 并利用第 4.3 小节的两阶段算法进行求解. 辅助加速度的罚参数  $\beta$  取  $5 \times 10^5$ . 控制第一和第二阶段终止条件的参数  $\eta$  和  $\epsilon$  分别取  $10^{-3}$  和  $10^{-6}$ . 在没有对飞行轨迹进行分析预测的情况下, 我们粗略地选取  $\tau_f = 5$ . 两阶段算法在 11 次迭代后终止. 终端质量和终端时刻分别为 13557.8 和 22.25. 所得飞行轨迹、质量、推力曲线和推力大小与变量  $\Gamma$  的关系如图 3-5 所示. 可以看到, 两阶段算法求解非凸最优控制问题 (ROC) 仅需求解 12 个 (包括初始化 1 个和两阶段共 11 个) SOCP 问题, 算法效率较高; 算法得到的结果满足推力上下限和推力方向等约束, 飞行轨迹较为平滑;  $\|\mathbf{T}_k\|$  与  $\Gamma_k$  在最终解处的距离不超过  $7 \times 10^{-8}$ , 所用无损凸化技巧确实达到“无损”的效果; 最终解的终端时刻与预测值相差 17.25 秒, 算法在如此粗糙的预测下仍能正常迭代并收敛.

表 1 测试问题数据及参数取值

参数	取值	单位
$N$	30	-
$\alpha$	$3.399 \times 10^{-4}$	秒/米
$\rho$	1	千克/米 <sup>3</sup>
$S_D$	10	米 <sup>2</sup>
$C_D$	1	-
$m_{\text{dry}}$	10000	千克
$\theta_1$	80	度
$\theta_2$	15	度
$T_{\text{min}}$	100000	牛顿
$T_{\text{max}}$	250000	牛顿
$\dot{T}_{\text{min}}$	-100000	牛顿/秒
$\dot{T}_{\text{max}}$	100000	牛顿/秒
$\mathbf{g}$	$[0 \ -9.807 \ 0]^T$	米/秒 <sup>2</sup>
$m_{\text{init}}$	15000	千克
$\mathbf{r}_{\text{init}}$	$[-600 \ 2000 \ 600]^T$	米
$\mathbf{v}_{\text{init}}$	$[80 \ -150 \ -80]^T$	米/秒

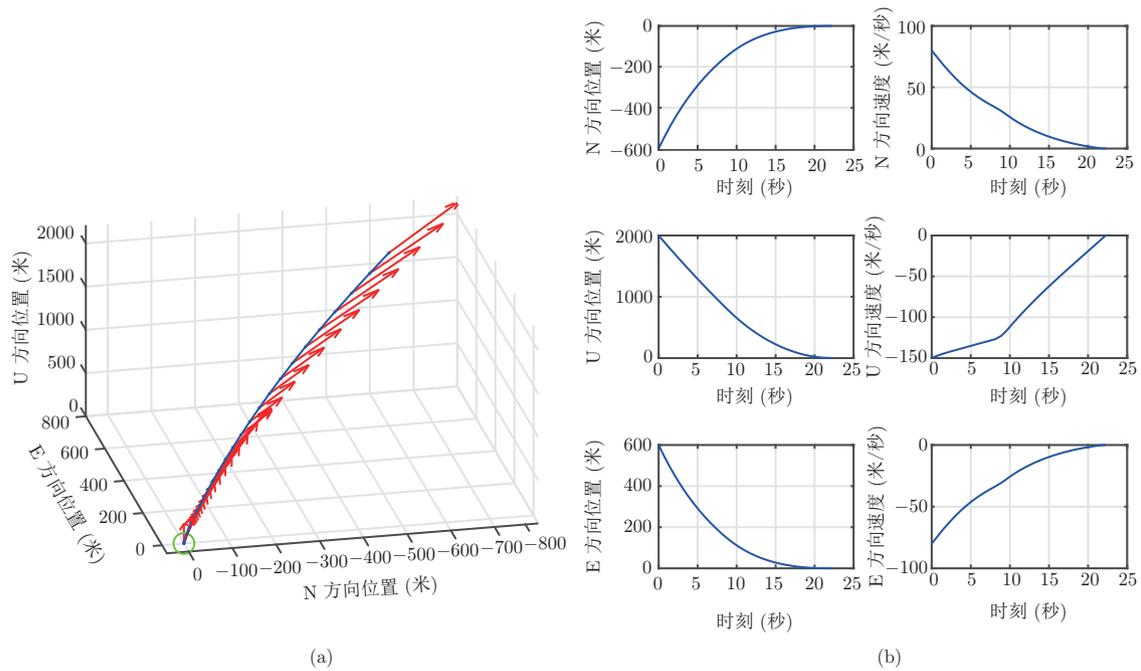


图 3 (网络版彩图) (a) 3D 位置轨迹, 其中箭头表示加速度, 圆形表示着陆点; (b) 2D 位置速度轨迹

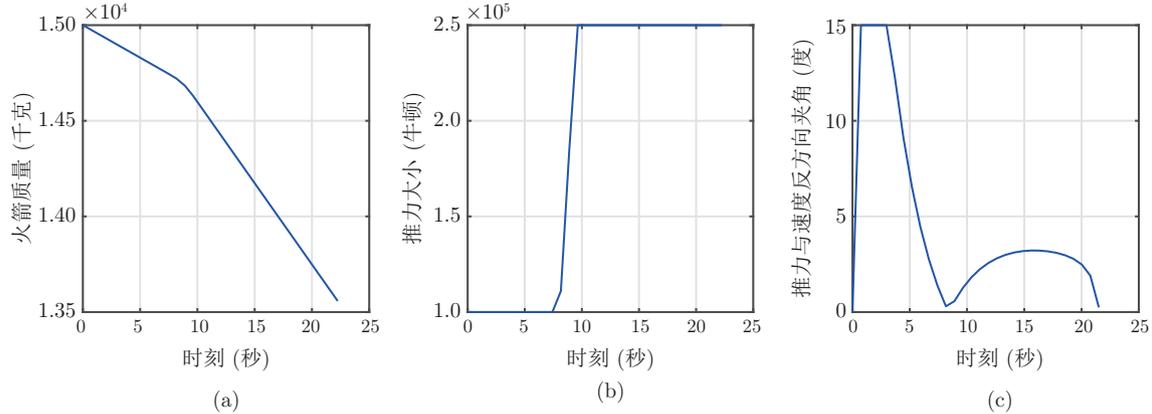


图 4 (网络版彩图) 质量 (a)、推力大小 (b) 和推力与速度反方向夹角曲线 (c)

为了进一步验证算法对  $\tau_f$  的弱依赖性, 同时展示两阶段方法相较于第 4.2 小节的单阶段序列凸化方法更加稳定和高效的特点, 我们选取不同的  $\tau_f$  进行多次测试. 图 6 展示了在 10 个不同的猜测值  $\tau_f$  下, 算法总可以使终端时刻  $t_f$  快速收敛 (6 次迭代之内), 且收敛到同一个值. 表 2 比较了单阶段方法与两阶段方法的迭代次数. 单阶段法在  $\tau_f$  取 10、15 和 25 时出现了数值不稳定性, 导致算法迭代次数异常增加. 相比之下, 两阶段法 10 次测试的迭代次数却可以控制在 11–13 次. 不仅在数值表现上更加稳定, 而且在单阶段法表现正常的测试中, 两阶段法也以更少的迭代次数胜出. 图 7 给出了与约束 (3.6) 相关的物理量曲线. 可以看到, 第一阶段所得的轨迹在经过第二阶段的迭代后满足了约束 (3.6), 但此轨迹与最终解差别很小. 由此可以猜测, 约束 (3.6) 比其他线性化约束对初始轨迹的质量更敏感.

两阶段方法中的第一阶段提供了很好的轨迹, 使得非凸约束 (2.7) 的序列凸化更加稳定. 因此, 两阶段方法具有比单阶段方法更好的稳定性.

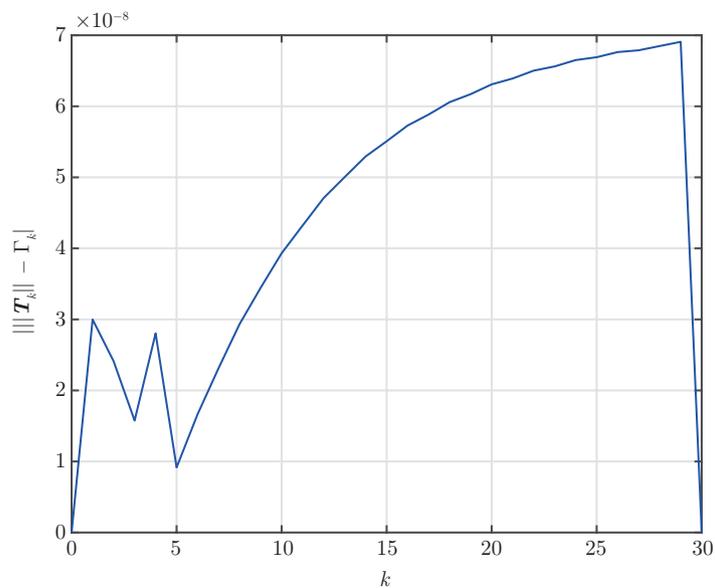


图 5 (网络版彩图) 最终解处实际推力大小  $\|T\|$  与推力上界  $\Gamma$  的关系

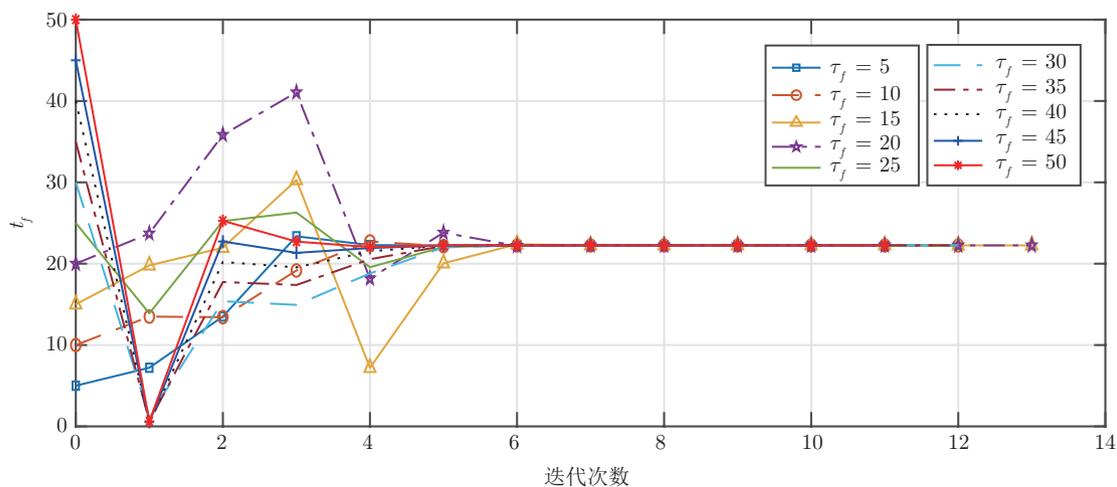


图 6 (网络版彩图) 不同终端时刻猜测值  $\tau_f$  下的  $t_f$  收敛结果

表 2 单阶段法和两阶段法的迭代次数对比

$\tau_f$	5	10	15	20	25	30	35	40	45	50
单阶段法	11	30	88	13	225	17	19	19	15	14
两阶段法	11	12	13	13	12	12	12	11	11	11

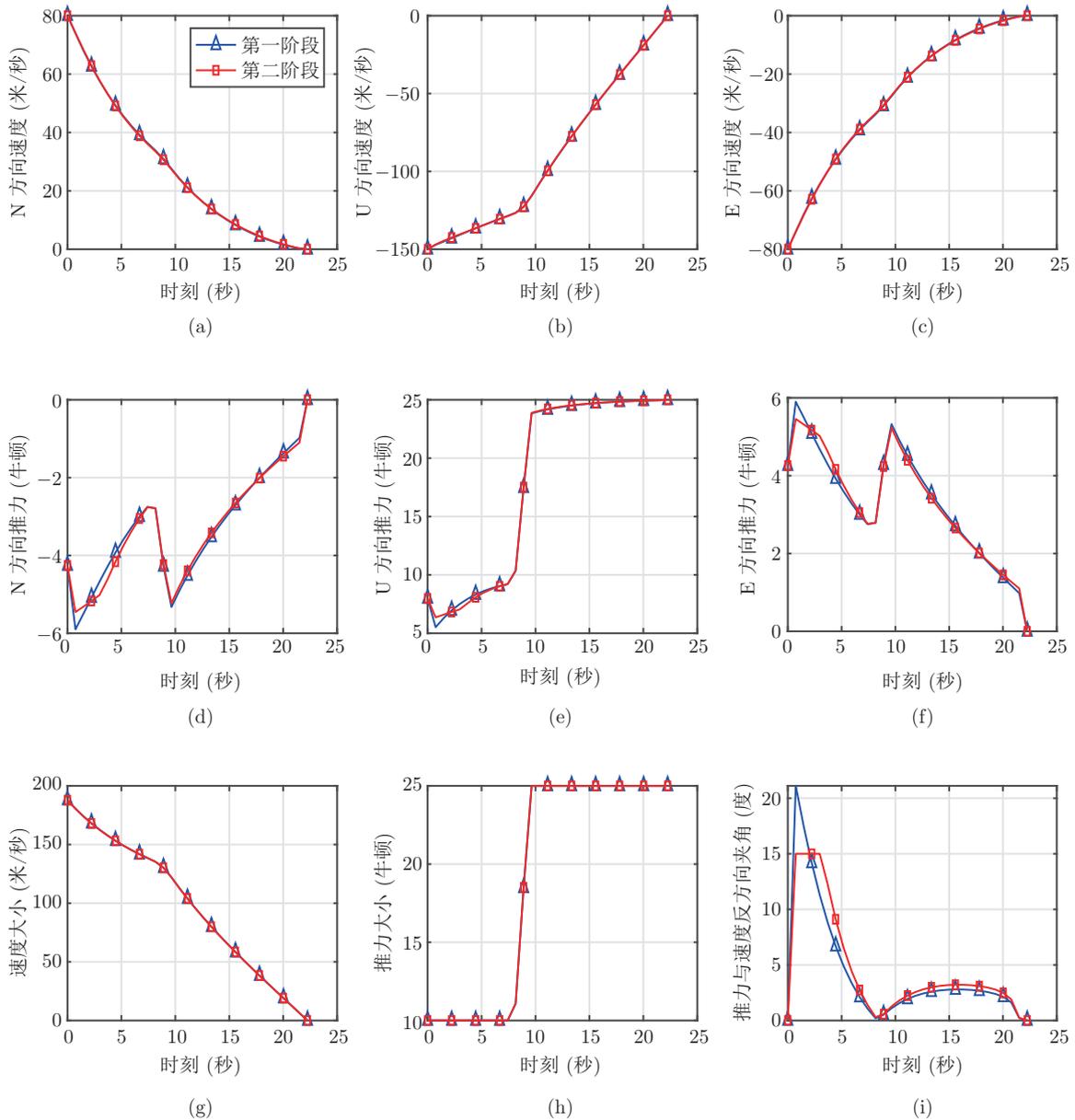


图 7 (网络版彩图) 各阶段最终解中的方向速度 (a)–(c)、方向推力 (d)–(f)、速度大小 (g)、推力大小 (h) 和推力与速度反方向夹角曲线 (i)

## 6 总结

本文考虑了包含非凸推力方向控制约束的 3-DoF 燃料最优动力下降制导最优控制问题. 通过无损凸化、时间离散化和线性化技术, 非凸最优控制问题被转化为 SOCP 问题进行求解. 为了避免数值不稳定的发生, 同时提高算法迭代效率, 我们在单阶段序列凸化方法的基础上提出了一个新的两阶段序列凸化方法. 数值结果表明所提方法可以通过求解少量的凸规化问题得到非凸最优控制问题的解,

并且算法的收敛性不依赖于初始猜测值的选取. 此外, 多个测试问题的结果均说明两阶段方法拥有比单阶段方法更稳定和高效的数值表现.

## 参考文献

- 1 Boyd S, Vandenberghe L. *Convex Optimization*. Cambridge: Cambridge University Press, 2004
- 2 Nesterov Y E, Todd M J. Self-scaled barriers and interior-point methods for convex programming. *Math Oper Res*, 1997, 22: 1–42
- 3 Nesterov Y, Nemirovskii A. *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia: SIAM, 1994
- 4 Sturm J F. Implementation of interior point methods for mixed semidefinite and second order cone optimization problems. *Optim Methods Softw*, 2002, 17: 1105–1154
- 5 Acikmese B, Ploen S R. Convex programming approach to powered descent guidance for Mars landing. *J Guidance Control Dyn*, 2007, 30: 1353–1366
- 6 Blackmore L, Açikmeşe B, Scharf D P. Minimum-landing-error powered-descent guidance for Mars landing using convex optimization. *J Guidance Control Dyn*, 2010, 33: 1161–1171
- 7 Carson J M, Acikmese B, Blackmore L. Lossless convexification of powered-descent guidance with non-convex thrust bound and pointing constraints. In: *Proceedings of the 2011 American Control Conference*. San Francisco: IEEE, 2011, 2651–2656
- 8 Acikmese B, Carson J M, Blackmore L. Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem. *IEEE Trans Contr Syst Technol*, 2013, 21: 2104–2113
- 9 Szmuk M, Acikmese B, Berning A W. Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints. In: *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. San Diego: American Institute of Aeronautics and Astronautics, 2016, 1–16
- 10 Szmuk M, Eren U, Acikmese B. Successive convexification for Mars 6-DoF powered descent landing guidance. In: *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. Grapevine: American Institute of Aeronautics and Astronautics, 2017, 1–15
- 11 Szmuk M, Acikmese B. Successive convexification for 6-DoF Mars rocket powered landing with free-final-time. In: *Proceedings of the 2018 AIAA Guidance, Navigation, and Control Conference*. Kissimmee: American Institute of Aeronautics and Astronautics, 2018, 1–14
- 12 Szmuk M, Reynolds T, Acikmese B, et al. Successive convexification for 6-DoF powered descent guidance with compound state-triggered constraints. In: *Proceedings of the AIAA Scitech 2019 Forum*. San Diego: American Institute of Aeronautics and Astronautics, 2019, 1–16
- 13 Liu X, Shen Z, Lu P. Solving the maximum-crossrange problem via successive second-order cone programming with a line search. *Aerosp Sci Tech*, 2015, 47: 10–20
- 14 Liu X, Shen Z, Lu P. Entry trajectory optimization by second-order cone programming. *J Guidance Control Dyn*, 2016, 39: 227–241
- 15 Wang Z, Grant M J. Autonomous entry guidance for hypersonic vehicles by convex optimization. *J Spacecraft Rockets*, 2018, 55: 993–1006
- 16 Lu P, Liu X. Autonomous trajectory planning for rendezvous and proximity operations by conic optimization. *J Guidance Control Dyn*, 2013, 36: 375–389
- 17 Szmuk M, Pascucci C A, Dueri D, et al. Convexification and real-time on-board optimization for agile quad-rotor maneuvering and obstacle avoidance. In: *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver: IEEE, 2017, 4862–4868
- 18 Zhang Z, Li J, Wang J. Sequential convex programming for nonlinear optimal control problems in UAV path planning. *Aerosp Sci Tech*, 2018, 76: 280–290
- 19 Liu X, Shen Z, Lu P. Exact convex relaxation for optimal flight of aerodynamically controlled missiles. *IEEE Trans Aerosp Electron Syst*, 2016, 52: 1881–1892
- 20 Grant M, Boyd S. *CVX: Matlab software for disciplined convex programming, version 2.1*. 2014
- 21 Grant M C, Boyd S P. Graph implementations for nonsmooth convex programs. In: *Proceedings of the Recent Advances in Learning and Control*. London: Springer, 2008, 95–110
- 22 Domahidi A, Chu E, Boyd S. ECOS: An SOCP solver for embedded systems. In: *Proceedings of the 2013 European Control Conference (ECC)*. Zürich: IEEE, 2013, 3071–3076

## A two-stage successive convexification method for the powered descent guidance problem

Muming Yang & Guo-You You

**Abstract** In recent five years, the reusable rocket technology is a focus topic in the aerospace engineering field. Aiming at the three-degree-of-freedom fuel-optimal control problem, this paper studies the key problem in the rocket recovery technology: the solution of the powered descent guidance problem. Direct methods on the optimal control problem can hardly meet the real-time requirement since the model contains nonconvex dynamics, including the aerodynamic drag term, and nonconvex thrust magnitudes and direction control constraints. To eliminate the difficulty coming from the nonconvexity, we transform the model to a kind of convex programming problem, i.e., a second order cone programming problem to solve by lossless convexification, time discretization, and linearization. In addition, this paper introduces a new two-stage successive convexification method to avoid the numerical unstable phenomenon. The numerical results show that the proposed method only needs to solve several convex optimization problems and is more stable and efficient than the traditional single stage method.

**Keywords** rocket, powered descent guidance, optimal control, convexification, two-stage

MSC(2010) 49J15, 90C25

doi: 10.1360/SSM-2020-0105