中国科学 : 物理学 力学 天文学 2025 年 第 55 卷 第 2 期 : 224519
SCIENTIA SINICA Physica, Mechanica & Astronomica physcn.scichina.com
《中国科学》杂志社 SCIENCE CHINA PRESS
论 文 先进航天器动力学与控制专辑I

针对受约束轨道追逃博弈问题的循环差分进化求解框架

张去非[1,2], 季明江[2], 闫斌[2], 刘才山[1*], 曹璐[2*]

1. 北京大学工学院, 北京 100871
2. 军事科学院国防科技创新研究院, 北京 100071
*联系人, 刘才山, E-mail: LIUCS@pku.edu.cn; 曹璐, E-mail: caolu_space2015@163.com

收稿日期: 2024-07-31; 接受日期: 2024-12-03; 网络出版日期: 2025-01-02

摘要 轨道追逃博弈问题得到了广泛的研究, 但大多数均建立在无过程约束的假设上. 本文针对受轨道高度约束的轨道追逃博弈问题, 推导了鞍点存在性条件. 基于方程系数矩阵的性质, 将其解耦降维为两组两点边值问题. 利用罚函数法处理过程约束, 并将两点边值问题转化为针对最优性条件的最优化问题. 继而提出了一种循环差分进化(DE)求解框架用于求解该最优化问题, 以兼顾全局搜索能力和收敛效率. 仿真结果表明, 该求解框架相较于传统差分进化算法, 大幅提高了求解效率和精度. 最后, 通过对比无轨道约束工况仿真结果, 揭示了轨道高度约束对于防止航天器与地球碰撞的现实意义.

关键词 轨道追逃博弈, 轨道高度约束, 罚函数, 两点边值问题, 循环DE求解框架

PACS: 02.60.Cb, 02.50.Le, 95.10.Eg, 02.30.Yy

1 引言

己方与非合作目标处于互斥立场, 通过实施控制达到纳什均衡, 被称作微分对策. 这一名词最早由Isaacs [1]提出并基于动态规划原理给出了对策值的最优性必要条件, 随后被应用到空战捕获, 共面追击-逃避以及轨道追逃博弈等领域[2–5]. 近年来, 航天器的轨道追逃博弈逐渐得到众多学者的关注. 典型的, 双边参与的航天器追逃博弈由一架追踪航天器和一架逃逸航天器构成, 并且双方具有排斥的任务目标: 追方和逃方分别希望通过机动缩短和延长交会的时间. 特别的, 当两者的定量目标函数和为零时, 则构成了轨道零和追逃博弈问题. Isaacs所提出的微分对策问题本质上就是一种零和博弈问题. 随后, 基于最优化理论[6], Başar和Olsder [7]讨论了鞍点开环表示的必要条件, 为开环鞍点的数值求解奠定了基础. 通过求解轨道零和追逃博弈的纳什均衡, 使双方都没有更好的机动选择, 构成了本文的研究主题.

很多研究关注于对轨道追逃博弈的纳什均衡的分析. 其中一部分研究将博弈空间划分为捕获区和逃逸区并研究其分界的界栅[8]; 更多的研究则是数值寻找轨道追逃博弈问题的鞍点. 一类直接方法将

引用格式: 张去非, 季明江, 闫斌, 等. 针对受约束轨道追逃博弈问题的循环差分进化求解框架. 中国科学: 物理学 力学 天文学, 2025, 55: 224519
Zhang Q F, Ji M J, Yan B, et al. A cyclic differential evolution framework for constrained orbital pursuit-evasion games (in Chinese). Sci Sin-Phys Mech Astron, 2025, 55: 224519, doi: 10.1360/SSPMA-2024-0310

© 2025 《中国科学》杂志社

双边优化问题转换为非线性规划任务, 这种方法被广泛使用. Raivio和Ehtamo [9]使用离散化和非线性规划技术解决了一个涉及视觉识别的追逃博弈问题. Hargraves和Paris [10]将嵌入式插值技术集成到他们的数学规划框架中. 而另一类间接方法则本质上等价于求解多点边界值问题(MPBVP), 并通常是两点边界值问题 [11,12]. 然而, 复杂系统导致的高维两点边值问题通常使轨道追逃博弈的鞍点求解极具挑战性 [13]. 为了解决这一难题, Pontani和Conway [14]引入了一种混合方法, 称为半直接方法, 该方法使用遗传算法为最优解生成初始猜测. 最近的研究大多基于该半直接方法, Li等人 [15]提出了一种降维技术, 用于高效地解决两点边值问题. 但这种降维技术一定程度上依赖于方程的线性性质. 为了克服这一限制, 采用打靶和插值的融合方法来求解非线性动力学系统中派生的两点边值子问题 [16]. 多段自适应伪谱法技术也被用于帮助生成初始猜测 [17]. Pang等人 [18]则基于线性的椭圆轨道方程, 利用解析梯度快速计算追逃博弈问题的鞍点. 除此以外, 部分学者利用深度学习和强化学习来解决此类轨道追逃博弈问题, 以便有效处理非合作目标具备复杂自主机动的情形 [19]. Tian等人 [20]提出了一种基于VAE的未授权频段频谱异常检测方法, 创新性地引入百分位评分来捕捉重构误差的变化, 总结了空间膜天线的发展历程及最新进展, 介绍了曲面反射器、共形有源膜天线、平面阵列膜天线和平面反射阵列膜天线, 涵盖了射频设计、展开机制、材料、实验、应用及分析方法, 并通过归纳当前膜天线的优势与挑战, 展望了空间膜天线的未来发展趋势 [21].

尽管轨道追逃博弈问题的数值求解得到了大量的研究, 但其中大部分都建立在理想的假设之上. 而在真实的太空作业环境中, 往往面临各方面的约束和限制, 譬如形状不规则小行星附近的轨道约束 [22,23]、地球扁率引起的J_2摄动影响 [24]、推力方向受约束 [25]以及不同卫星轨道运行高度限制等.

因此, 为了使轨道追逃博弈问题的均衡解更具有实际意义, 本文研究了航天器受轨道高度约束下的轨道追逃博弈问题. 首先推导了该问题的最优性条件, 本质上对应着一组高维的两点边值问题. 基于方程系数矩阵的性质, 简化了该两点边值问题并转化成为了一个最优化问题. 随后, 利用罚函数法, 将航天器所受过程约束等效为需要优化的目标函数, 并嵌入此最优化问题中. 最后, 提出了一种循环差分进化(DE)求解框架, 具备较强的全局搜索能力, 并且较于传统DE算法, 该框架具备更高的求解效率和精度.

本文的剩余部分分为如下几个部分: 第2节, 阐述了零和轨道微分追逃博弈问题的动力学模型和鞍点存在的最优性条件; 第3节, 提供了本文的求解框架的具体细节; 第4节, 对受轨道约束的轨道追逃博弈问题进行了仿真, 展示了仿真结果, 并讨论了算法的效率和轨道约束的影响; 第5节, 总结了本篇论文.

2 预备知识

本节中, 航天器的动力学模型将建立在地心笛卡儿坐标系中, 基于该坐标系, 构建一对一零和航天器追逃博弈模型, 并推导零和轨道追逃博弈的鞍点存在条件.

2.1 非线性动力学模型

如图1所示, 在地心建立笛卡儿坐标系O-xyz, 以$\boldsymbol{X}_i = [x_i, y_i, z_i, v_{xi}, v_{yi}, v_{zi}]^{\mathrm{T}}$描述航天器状态, 最大推力比视为常量$T_i$, 以$u_i$描述航天器推力幅值占最大推力比的比例, 以$\boldsymbol{\alpha}_i$描述航天器推力方向的单位矢量, 以$\boldsymbol{f}_i$描述航天器状态的导函数. 此外, 为方便表示, 令$i = P, E$, 其中P代表追方航天器, E代表逃方航天器, 则航天器的动力学模型可建立如下:

$$\dot{\boldsymbol{X}}_i = \boldsymbol{f}_i(\boldsymbol{X}_i, u_i, \hat{\boldsymbol{\alpha}}_i), \tag{1}$$

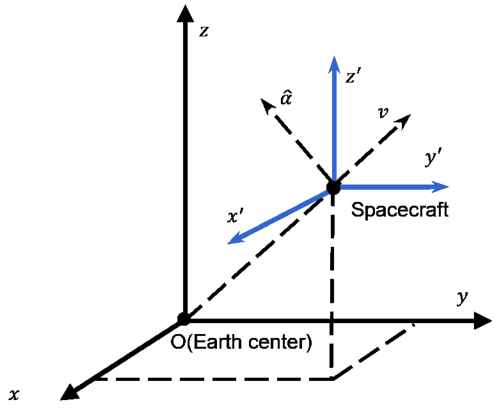

图 1 (网络版彩图)全局坐标系

Figure 1 (Color online) Cartesian coordinate system.

其中,

$\hat{\boldsymbol{\alpha}}_i = [\alpha_{xi}, \alpha_{yi}, \alpha_{zi}]^{\mathrm{T}}$,

$u_i \in [0, 1]$.

动力学方程表述如下:

$$
\begin{aligned}
\dot{x}_i &= v_{xi},\\
\dot{y}_i &= v_{yi},\\
\dot{z}_i &= v_{zi},\\
\dot{v}_{xi} &= T_i u_i \alpha_{xi} - \frac{\mu_E}{x_i^2 + y_i^2 + z_i^2}\beta_{xi},\\
\dot{v}_{yi} &= T_i u_i \alpha_{yi} - \frac{\mu_E}{x_i^2 + y_i^2 + z_i^2}\beta_{yi},\\
\dot{v}_{zi} &= T_i u_i \alpha_{zi} - \frac{\mu_E}{x_i^2 + y_i^2 + z_i^2}\beta_{zi},
\end{aligned}
\tag{2}
$$

其中, $\mu_E = 3.986004418 \times 10^{14}\ \mathrm{m^3/s^2}$为地心引力常数, $\boldsymbol{\beta}_i = [\beta_{xi}, \beta_{yi}, \beta_{zi}]^{\mathrm{T}}$表示坐标轴方向的引力分量:

$$
\beta_{ji} = \frac{j_i}{\sqrt{x_i^2 + y_i^2 + z_i^2}}, \quad j = x, y, z.
$$

如图2所示, 推力单位矢量$\hat{\boldsymbol{\alpha}}_i$可以利用在局部坐标系O′-ENR中定义的两个方向角表示. 其中N表示沿经线向北, 而E表示沿经线向东. 推力单位矢量$\hat{\boldsymbol{\alpha}}_i$与方向角$\theta_i$及$\psi_i$存在如下转换关系:

$$
\begin{aligned}
\alpha_{xi} =& -\cos\psi_i\cos\theta_i\frac{y_i}{\sqrt{x_i^2 + y_i^2}}\\
&-\cos\psi_i\sin\theta_i\frac{x_i z_i}{\sqrt{x_i^2 + y_i^2}\sqrt{x_i^2 + y_i^2 + z_i^2}}\\
&+\sin\psi_i\frac{x_i}{\sqrt{x_i^2 + y_i^2 + z_i^2}},\\
\alpha_{yi} =& \cos\psi_i\cos\theta_i\frac{x_i}{\sqrt{x_i^2 + y_i^2}}\\
&-\cos\psi_i\sin\theta_i\frac{y_i z_i}{\sqrt{x_i^2 + y_i^2}\sqrt{x_i^2 + y_i^2 + z_i^2}}\\
&+\sin\psi_i\frac{y_i}{\sqrt{x_i^2 + y_i^2 + z_i^2}},\\
\alpha_{zi} =& \cos\psi_i\sin\theta_i\frac{\sqrt{x_i^2 + y_i^2}}{\sqrt{x_i^2 + y_i^2 + z_i^2}}\\
&+\sin\psi_i\frac{z_i}{\sqrt{x_i^2 + y_i^2 + z_i^2}}.
\end{aligned}
$$

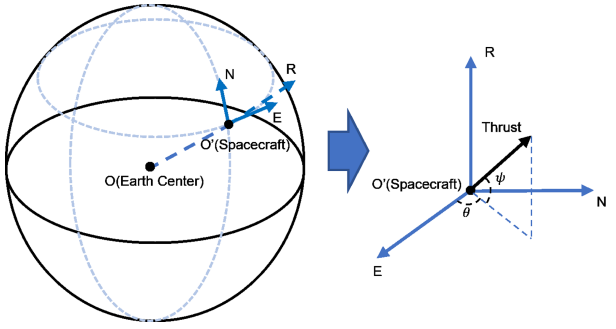

图 2 (网络版彩图)局部坐标系

Figure 2 (Color online) Local coordinate system.

方向角θ_i及ψ_i分别揭示了航天器经纬坐标和轨道高度的变化趋势, 一定程度上反映了追逃双方在博弈过程中采取的策略.

为了描述双方参与的动力学系统, 将状态量拓维至12维, 则整个博弈系统的动力学可表达为

$$
\dot{\boldsymbol{X}} = \boldsymbol{F} = [\boldsymbol{f}_p(\boldsymbol{X}_p, u_p, \hat{\boldsymbol{\alpha}}_p, t)^{\mathrm{T}}, \boldsymbol{f}_e(\boldsymbol{X}_e, u_e, \hat{\boldsymbol{\alpha}}_e, t)^{\mathrm{T}}]^{\mathrm{T}},
\tag{3}
$$

其中, $\boldsymbol{X}$由追踪者的状态$\boldsymbol{X}_p$和逃逸者的状态$\boldsymbol{X}_e$组成, 即$\boldsymbol{X} = [\boldsymbol{X}_p^{\mathrm{T}}, \boldsymbol{X}_e^{\mathrm{T}}]^{\mathrm{T}}$.

2.2 双边零和博弈

双方参与的零和博弈中, 参与的双方拥有互斥的定量目标J_p和J_e, 同时, $J_p + J_e$为零或常数. 本文假设双方参与者仅考虑博弈持续时间, 追方希望缩短博弈时间, 而逃方则希望延长博弈时间, 则最终的支付函数可表达为

$$
\begin{aligned}
J_p &= t_f,\\
J_e &= -t_f.
\end{aligned}
\tag{4}
$$

在整个博弈过程中, 追逃双方的轨道高度受到限制:

$$
h_i \in [\mathrm{Low}_i + R_E, \mathrm{Upper}_i + R_E],
\tag{5}
$$

其中, $h_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$表示航天器$i$的地心距离, Low_i和Upper_i分别表示航天器i轨道高度的下限和上限, $R_E = 6378137\ \mathrm{m}$表示地球半径.

双方在$t = 0$时刻状态的初始值为

$$
\boldsymbol{X}_p(0) = \boldsymbol{X}_{p0}, \quad \boldsymbol{X}_e(0) = \boldsymbol{X}_{e0}.
\tag{6}
$$

博弈终端时间t_f由预定义的终止集决定:

$$\boldsymbol{\phi} = [x_p - x_e, y_p - y_e, z_p - z_e]^{\mathrm{T}}|_{t_f} = \mathbf{0}_{3\times 1}. \tag{7}$$

博弈达到了一个纳什均衡, 当且仅当下列不等式满足:

$$\begin{aligned} &J_e(u_p^*, \hat{\boldsymbol{\alpha}}_p^*, u_e, \hat{\boldsymbol{\alpha}}_e) \geq J_e(u_p^*, \hat{\boldsymbol{\alpha}}_p^*, u_e^*, \hat{\boldsymbol{\alpha}}_e^*), \\ &J_p(u_p, \hat{\boldsymbol{\alpha}}_p, u_e^*, \hat{\boldsymbol{\alpha}}_e^*) \geq J_p(u_p^*, \hat{\boldsymbol{\alpha}}_p^*, u_e^*, \hat{\boldsymbol{\alpha}}_e^*). \end{aligned} \tag{8}$$

2.3 最优性条件

对于每一个参与者, 引入协态变量来构建对应的哈密顿函数:

$$H_i = \boldsymbol{\Lambda}_i^{\mathrm{T}} \boldsymbol{F}, \tag{9}$$

其中, 协态变量$\boldsymbol{\Lambda}_i$的维度为12×1.

根据庞特里亚金极大值原理[26], 最优控制输入$\{u_i, \boldsymbol{\alpha}_i\}$最小化哈密顿量$H_i$:

$$\begin{aligned} [u_p^*, \hat{\boldsymbol{\alpha}}_p^*] &= \underset{u_p, \hat{\boldsymbol{\alpha}}_p}{\operatorname{argmin}} H_p = \underset{u_p, \hat{\boldsymbol{\alpha}}_p}{\operatorname{argmin}} \Lambda_{p4} T_p u_p \alpha_{xp} \\ &\quad + \Lambda_{p5} T_p u_p \alpha_{yp} + \Lambda_{p6} T_p u_p \alpha_{zp}, \\ [u_e^*, \hat{\boldsymbol{\alpha}}_e^*] &= \underset{u_e, \hat{\boldsymbol{\alpha}}_e}{\operatorname{argmin}} H_e = \underset{u_e, \hat{\boldsymbol{\alpha}}_e}{\operatorname{argmin}} \Lambda_{e10} T_e u_e \alpha_{xe} \\ &\quad + \Lambda_{e11} T_e u_e \alpha_{ye} + \Lambda_{e12} T_e u_e \alpha_{ze}. \end{aligned} \tag{10}$$

于是, 控制量可由式(10)推出

$$\boldsymbol{\alpha}_i^* = -\frac{\boldsymbol{\Lambda}_{vi}}{\|\boldsymbol{\Lambda}_{vi}\|}, \qquad u_i^* = 1. \tag{11}$$

此外, 最优性条件包括下列协态变量方程:

$$\dot{\boldsymbol{\Lambda}}_i = -\frac{\partial Hi}{\partial \boldsymbol{X}} = -\left[\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}}\right]^{\mathrm{T}} \boldsymbol{\Lambda}_i, \tag{12}$$

其中,

$$\left[\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}}\right]^{\mathrm{T}} = \begin{bmatrix} \mathbf{0}_{3\times 3} & \boldsymbol{G}_p & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \boldsymbol{I}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \boldsymbol{G}_e \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \boldsymbol{I}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix}. \tag{13}$$

在式(13)中

$$\boldsymbol{G}_i = c \begin{bmatrix} 2x_i^2 - y_i^2 - z_i^2 & 3x_i y_i & 3x_i z_i \\ 3x_i y_i & 2y_i^2 - x_i^2 - z_i^2 & 3y_i z_i \\ 3x_i z_i & 3y_i z_i & 2z_i^2 - x_i^2 - y_i^2 \end{bmatrix},$$

$$c = \frac{\mu_E}{(x_i^2 + y_i^2 + z_i^2)^{\frac{5}{2}}}.$$

协态变量的终端边界条件如下所示:

$$\begin{aligned} \boldsymbol{\Lambda}_p(t_f) &= \frac{\partial\left(t_f + \boldsymbol{\upsilon}_p^{\mathrm{T}}\boldsymbol{\phi}\right)}{\partial \boldsymbol{X}}|_{t=t_f} = [\boldsymbol{\upsilon}_p^{\mathrm{T}}, \mathbf{0}_{1\times 3}, -\boldsymbol{\upsilon}_p^{\mathrm{T}}, \mathbf{0}_{1\times 3}]^{\mathrm{T}}, \\ \boldsymbol{\Lambda}_e(t_f) &= \frac{\partial\left(-t_f + \boldsymbol{\upsilon}_e^{\mathrm{T}}\boldsymbol{\phi}\right)}{\partial \boldsymbol{X}}|_{t=t_f} = [\boldsymbol{\upsilon}_e^{\mathrm{T}}, \mathbf{0}_{1\times 3}, -\boldsymbol{\upsilon}_e^{\mathrm{T}}, \mathbf{0}_{1\times 3}]^{\mathrm{T}}. \end{aligned} \tag{14}$$

其中, $\boldsymbol{\upsilon}_i = [\upsilon_{i1}, \upsilon_{i2}, \upsilon_{i3}]^{\mathrm{T}}$表示拉格朗日乘子向量.

最后, 哈密顿量的终端约束如下所示:

$$\begin{aligned} H_p(t_f) &= -\frac{\partial\left(t_f + \boldsymbol{\upsilon}_p^{\mathrm{T}}\boldsymbol{\phi}\right)}{\partial t_f} = -1, \\ H_e(t_f) &= -\frac{\partial\left(-t_f + \boldsymbol{\upsilon}_e^{\mathrm{T}}\boldsymbol{\phi}\right)}{\partial t_f} = 1. \end{aligned} \tag{15}$$

式(15)表明存在一组特解满足

$$\boldsymbol{\upsilon}_p = -\boldsymbol{\upsilon}_e. \tag{16}$$

综上所述, 式(3), (6), (7)和(12)–(16)共同形成37维两点边值问题(TPBVP). 通过解决这个问题, 可以进一步计算空间运动轨迹和控制轨迹.

3 求解方法

本节分3个部分阐述本篇文章求解带状态约束的轨道追逃博弈问题所采用的方法: (1) 根据推导得出的37维TPBVP的性质, 对方程进行解耦降维, 最终得到两组边界耦合的12维TPBVP; (2) 对状态约束进行光滑化处理, 利用罚函数法以保证博弈过程中航天器对状态约束的满足; (3) 将TPBVP转化为最优化问题, 构建基于DE算法的循环求解框架, 以实现高效率高精度求解.

3.1 TPBVP解耦降维

高维TPBVP会导致计算量几何级地增长. 幸运的是, 因为方程的对称性, 该问题可以进行简化. 首先, 由式(10)和(11)可得, 双方的控制分别仅仅受到Λ_{vp}和Λ_{ve}的影响. 其次, 由于式(13)中协态方程的系数矩阵具有分块对角性质, 双方的协态变量前六维和后六维是解耦的, 这意味着Λ_{vp}的后六维及Λ_{ve}的前六维对状态转移没有影响. 最后, 由于本身系统的动力学方程是通过扩展状态变量得到, 追逃双方的动力学方

程也是解耦的. 综上所述, 37维的TPBVP可以降维至两组存在边界耦合的12维TPBVP, 如下所示:

$$\dot{\boldsymbol{y}}_p = \boldsymbol{F}_p(\boldsymbol{y}_p) \Longrightarrow \begin{cases} \dot{\boldsymbol{X}}_p = \boldsymbol{f}_p(\boldsymbol{X}_p, u_p, \boldsymbol{\alpha}_p), \\ \dot{\boldsymbol{\lambda}}_p = \left[\dfrac{\partial \boldsymbol{f}_p(\boldsymbol{X}_p, u_p, \boldsymbol{\alpha}_p)}{\partial \boldsymbol{X}_p}\right]^{\mathrm{T}} \boldsymbol{\lambda}_p, \end{cases} \tag{17}$$

$$\dot{\boldsymbol{y}}_e = \boldsymbol{F}_e(\boldsymbol{y}_e) \Longrightarrow \begin{cases} \dot{\boldsymbol{X}}_e = \boldsymbol{f}_e(\boldsymbol{X}_e, u_e, \boldsymbol{\alpha}_e), \\ \dot{\boldsymbol{\lambda}}_e = \left[\dfrac{\partial \boldsymbol{f}_e(\boldsymbol{X}_e, u_e, \boldsymbol{\alpha}_e)}{\partial \boldsymbol{X}_e}\right]^{\mathrm{T}} \boldsymbol{\lambda}_e, \end{cases} \tag{18}$$

其中, $\boldsymbol{\lambda}_p = [\Lambda_{p1}, \Lambda_{p2}, \Lambda_{p3}, \Lambda_{p4}, \Lambda_{p5}, \Lambda_{p6}]^{\mathrm{T}}$, $\boldsymbol{\lambda}_e = [\Lambda_{e7}, \Lambda_{e8}, \Lambda_{e9}, \Lambda_{e10}, \Lambda_{e11}, \Lambda_{e12}]^{\mathrm{T}}$, 同时定义新的状态变量$\boldsymbol{y}_i = [\boldsymbol{X}_i^{\mathrm{T}}, \boldsymbol{\lambda}_i^{\mathrm{T}}]^{\mathrm{T}}$.

3.2 状态约束处理

如式(5)所示, 由于双方航天器存在轨道高度上的状态约束, 3.1节所推导出的方程组得出的最优解并不一定满足该约束. 在本小节中, 首先对航天器所受的非线性轨道约束进行光滑化处理, 然后通过该光滑函数改写系统的状态转移方程, 以发挥罚函数对系统状态转移的影响作用.

重新整理式(5)中的不等式约束得

$$\begin{aligned} g_{i1} &= \mathrm{Low}_i - h_i \le 0, \\ g_{i2} &= h_i - \mathrm{Upper}_i \le 0. \end{aligned} \tag{19}$$

本文所采用的光滑技术由国内学者在2004年提出[27], 追逃双方分别存在两条不等式约束, 需要分别生成对应的光滑函数:

$$S_{i1} = \begin{cases} 0, & g_{i1} < -\dfrac{\epsilon}{\rho}, \\ \dfrac{\rho}{2\epsilon} g_{i1}^2 + g_{i1} + \dfrac{\epsilon}{2\rho}, & -\dfrac{\epsilon}{\rho} \le g_{i1} < 0, \\ g_{i1} + \dfrac{\epsilon}{2\rho}, & g_{i1} > 0, \end{cases}$$

$$S_{i2} = \begin{cases} 0, & g_{i2} < -\dfrac{\epsilon}{\rho}, \\ \dfrac{\rho}{2\epsilon} g_{i2}^2 + g_{i2} + \dfrac{\epsilon}{2\rho}, & -\dfrac{\epsilon}{\rho} \le g_{i2} < 0, \\ g_{i2} + \dfrac{\epsilon}{2\rho}, & g_{i2} > 0, \end{cases} \tag{20}$$

其中, ρ和ϵ是光滑因子.

通过对轨道约束作光滑化处理, 将过程约束等价为需要优化的目标函数, 并嵌入航天器的原支付函数中. 同时, 式(20)有效避免了在生成最优性条件时, 原过程约束不可微带来的困难.

将上述光滑函数作为惩罚项添加到支付函数中得到

$$\overline{J}_i = J_i + \rho \int_0^{t_f} (S_{i1} + S_{i2})\, \mathrm{d}t. \tag{21}$$

这将最终使得式(17)和(18)中的协态方程发生变化:

$$\begin{aligned} \dot{\boldsymbol{\lambda}}_p &= \left[\frac{\partial \boldsymbol{f}_p(\boldsymbol{X}_p, u_p, \boldsymbol{\alpha}_p)}{\partial \boldsymbol{X}_p}\right]^{\mathrm{T}} \boldsymbol{\lambda}_p + \frac{\partial(\rho(S_{p1} + S_{p2}))}{\partial \boldsymbol{X}_p}, \\ \dot{\boldsymbol{\lambda}}_e &= \left[\frac{\partial \boldsymbol{f}_e(\boldsymbol{X}_e, u_e, \boldsymbol{\alpha}_e)}{\partial \boldsymbol{X}_e}\right]^{\mathrm{T}} \boldsymbol{\lambda}_e + \frac{\partial(\rho(S_{e1} + S_{e2}))}{\partial \boldsymbol{X}_e}. \end{aligned} \tag{22}$$

3.3 循环DE求解框架

尽管3.1节推导出的两组TPBVP是独立的, 但他们拥有耦合的边界条件. 为了方便后续表述, 首先定义:

$$\begin{aligned} &\boldsymbol{p}_i = [x_i, y_i, z_i]^{\mathrm{T}}, \\ &\boldsymbol{v}_i = [v_{xi}, v_{yi}, v_{zi}]^{\mathrm{T}}, \\ &\boldsymbol{\lambda}_{posi} = [\lambda_{i1}, \lambda_{i2}, \lambda_{i3}]^{\mathrm{T}}, \\ &\boldsymbol{\lambda}_{vi} = [\lambda_{i4}, \lambda_{i5}, \lambda_{i6}]^{\mathrm{T}}. \end{aligned}$$

为了解决这个问题, 将两组TPBVP转换为针对边界初值的最优化问题.

$$\begin{aligned} &\min_{\boldsymbol{\lambda}_p(0), \boldsymbol{\lambda}_e(0)} |C_{eqj}|, \quad j = 1, 2, \ldots, 13, \\ &\boldsymbol{C}_{eq} = [\boldsymbol{C}_x, \boldsymbol{C}_\lambda, C_H]^{\mathrm{T}}, \end{aligned} \tag{23}$$

其中,

$$\begin{aligned} &\boldsymbol{C}_x = \boldsymbol{\phi}^{\mathrm{T}}, \\ &\boldsymbol{C}_\lambda = [\boldsymbol{\lambda}_{posp}^{\mathrm{T}} - \boldsymbol{\lambda}_{pose}^{\mathrm{T}}, \boldsymbol{\lambda}_{vp}^{\mathrm{T}}, \boldsymbol{\lambda}_{ve}^{\mathrm{T}}]|_{t_f}, \\ &\boldsymbol{C}_H = [\boldsymbol{\lambda}_p^{\mathrm{T}}(\boldsymbol{f}_p - \boldsymbol{f}_e) + 1]|_{t_f}. \end{aligned} \tag{24}$$

通过该最优化过程, 寻找两组TPBVP的边界初值, 将其转化为易于求解的初值问题(IVP). 因此该最优化过程的优化变量为两组TPBVP的边界初值, 优化目标为零和轨道博弈的最优性条件. 式(24)中, $\boldsymbol{C}_x$表示交会条件, 与式(7)一致; $\boldsymbol{C}_\lambda$表示协态变量的边界条件, 与式(14)一致; $\boldsymbol{C}_H$表示哈密顿函数的终端条件, 在$t = t_f$时, 协态变量满足式(14), 将式(14)代入式(9)和(15)中, 得到关于哈密顿函数的两个最优终端条件, 再根据式(16), 两个最优终端条件可以统一为$\boldsymbol{C}_H$.

此处的最优化目的是寻找使得$\boldsymbol{C}_{eq}$最接近于$\boldsymbol{0}_{13\times1}$的$t=0$时刻的$\boldsymbol{\lambda}_p$和$\boldsymbol{\lambda}_e$, 以及博弈持续时间$t_f$.

因此, 在进行DE算法时, 博弈时间t_f和$t=0$时刻的$\boldsymbol{\lambda}_p$和$\boldsymbol{\lambda}_e$被视为优化参数:

$$\{\boldsymbol{\lambda}_p|_{t=0}, \boldsymbol{\lambda}_e|_{t=0}, t_f\}. \tag{25}$$

为了便于处理该多目标优化问题, 13维的$\boldsymbol{C}_{eq}$被转化成为一个正定二次型的标量:

$$J=\boldsymbol{C}_{eq}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{C}_{eq}, \tag{26}$$

其中, $\boldsymbol{W}\in\mathbb{R}^{13\times13}$是需要设计的权值矩阵.

该最优化问题的求解流程如下: 首先, 如式(10)和(11)所示, 控制输入由协态变量决定. 随后, 分别从0到t_f 积分得到状态变量和协态变量轨迹. 随后, 基于终端时刻追方航天器和逃方航天器的状态变量$\boldsymbol{X}_i$和协态变量$\boldsymbol{\lambda}_i$, 计算得到目标函数J的值, 并通过评估J以此指导优化参数的更新.

为了求解上述最优化问题, 本文使用了DE算法, 这是一种基于种群的随机优化算法, 特别适用于求解非线性、非凸性问题. DE 算法的优势在于其具备全局搜索能力, 并能够在复杂搜索空间中找到全局最优解, 而不仅仅是局部最优解. 本文优化问题具有较高的非线性特征, 目标函数表面可能存在多个局部极小值, 并且大量分布于最优解小的邻域内. 因此, 传统的优化算法容易陷入局部最优解, 而难以找到全局最优. 为了解决这一问题, 在DE算法的基础上进行改进, 并采用了自适应变异策略以增强其全局搜索能力.

尽管DE算法具有全局搜索能力, 但其变异策略依赖于固定的缩放因子, 在搜索不同区域时, 容易导致算法过早陷入局部最优或无法充分探索搜索空间. 为了解决这个问题, 本文引入一种自适应变异策略[28]. 在该策略下, 缩放因子会根据种群的进化情况自适应调整, 基于每一代种群的适应度值变化, 动态调节缩放因子的取值范围. 当种群趋于收敛时, 降低缩放因子以提高局部搜索能力; 而当种群分散时, 增大缩放因子以增强全局搜索能力, 从而有效跳出局部最优解. 尽管如此, 改进DE算法仍然容易在某个局部极小值点收敛, 因此本文构建了一种循环DE求解框架, 通过循环迭代, 逐步缩小搜索空间并精细化搜索过程.

循环DE求解框架首先使用标准的改进DE算法对整个参数空间进行全局搜索, 获取第一个循环的最优解. 其第k次循环的DE算法的搜索范围基于第$k-1$次循环的DE算法搜索得到的最优参数, 并且随着循环次数的增加, 搜索半径逐渐缩小, 搜索空间逐渐收敛到全局最优解附近. 当连续若干次循环中搜索所得的最优解变化小于某一阈值时, 算法终止, 输出最终最优解. 算法流程如Algorithm 1所示.

Algorithm 1 循环DE求解框架

Require: 误差容限ε,
初始搜索范围SR
1: 基于SR, 通过改进差分进化算法获得初始猜测IG和支付函数J;
2: **while** $J\geq\varepsilon$ **do**
3: 更新SR至IG的邻域SR^+;
4: 基于SR^+, 通过改进差分进化算法更新获得初始猜测IG^+和支付函数J_+;
5: **if** $J^+\leq J$ **then**
6: 用J^+替代J , IG^+替代IG, SR^+替代SR;
7: **end if**
8: **end while**
9: **return** 初始猜测$\{\lambda_P, \lambda_E, t_f\}$, 满足条件$J\leq\varepsilon$.

4 结果和讨论

本节借鉴文献[29]工况, 利用循环DE求解框架对零和轨道追逃博弈鞍点进行求解, 通过对比展示该算法的精度和效率方面的优势; 随后, 将计算结果与未考虑过程约束的零和轨道追逃博弈进行对比, 展示了轨道高度约束的科学性和实际意义.

4.1 实验设置

在进行仿真之前, 为了更方便计算, 对时间单位和长度单位进行无量纲化处理:

$$DU=R_E,\quad TU=\sqrt{\frac{R_E^3}{\mu_E}}.$$

其他物理量的单位可由DU和TU导出.

令

$$\boldsymbol{W}=10\times\mathrm{diag}(100,100,100,1,1,1,1,1,1,1,1,1,1).$$

在利用循环DE算法框架求解的过程中, 第一次的全局搜索强调更大的搜索范围, 而其余的全局搜索范围则基于上一次最优值的邻域, 其范围与误差量级相

关. 所以自适应调节的搜索范围如下所示:

$$lb_j^{(1)} = \begin{cases} -15, & j = 1, \cdots, 12, \\ 1.239, & j = 13, \end{cases}$$

$$ub_j^{(1)} = \begin{cases} 15, & j = 1, \cdots, 12, \\ 12.39, & j = 13, \end{cases}$$

$$lb_j^{(k)} = \begin{cases} bv_j^{(k-1)} - \dfrac{\mathrm{err}}{2000}, & j = 1, \cdots, 12, \quad k = 2, 3, \cdots, \\ bv_j^{(k-1)} - \dfrac{\mathrm{err}}{6000}, & j = 13, \quad k = 2, 3, \cdots, \end{cases}$$

$$ub_j^{(k)} = \begin{cases} bv_j^{(k-1)} + \dfrac{\mathrm{err}}{2000}, & j = 1, \cdots, 12, \quad k = 2, 3, \cdots, \\ bv_j^{(k-1)} + \dfrac{\mathrm{err}}{6000}, & j = 13, \quad k = 2, 3, \cdots, \end{cases}$$

其中, $lb_j^{(k)}$和$ub_j^{(k)}$分别代表第k次迭代第j个参数的上界和下界.

此外, 改进DE算法的最大搜索次数和种群数量设置如下:

$$SN^{(k)} = \begin{cases} 200, & k = 1, \\ 100, & k = 2, 3, \cdots, \end{cases}$$

$$PS^{(k)} = \begin{cases} 200, & k = 1, \\ 100, & k = 2, 3, \cdots. \end{cases}$$

在仿真过程中, 使用了Matlab ode45和parallel computing toolbox, 仿真使用的硬件环境为Intel i5-13600 K 3.50 GHz和32 GB内存.

4.2 仿真结果与分析

仿真选用了文献[29]中第三章B节的工况, 其初始状态设置如下:

$$P \Rightarrow \begin{cases} x_{p0} = 6478.228\,\mathrm{km}, \\ y_{p0} = 1142.286\,\mathrm{km}, \\ z_{p0} = 0\,\mathrm{km}, \\ v_{xp0} = -1.171\,\mathrm{km/s}, \\ v_{yp0} = 6.639\,\mathrm{km/s}, \\ v_{zp0} = 3.892\,\mathrm{km/s}, \end{cases} \quad E \Rightarrow \begin{cases} x_{p0} = 5243.186\,\mathrm{km}, \\ y_{p0} = 3874.991\,\mathrm{km}, \\ z_{p0} = 875.402\,\mathrm{km}, \\ v_{xp0} = -3.634\,\mathrm{km/s}, \\ v_{yp0} = 3.592\,\mathrm{km/s}, \\ v_{zp0} = 5.872\,\mathrm{km/s}. \end{cases}$$

在仿真过程中, 设置轨道高度约束以避免航天器与地球碰撞, 即设置式(5)中Low_i及Upper_i如下:

$$\begin{aligned} &\mathrm{Low}_i = R_E + 100\,\mathrm{km}. \\ &\mathrm{Upper}_i = R_E + 2000\,\mathrm{km}. \end{aligned} \tag{27}$$

与文献[29]不同, 在轨道高度受限下, 为保证双方有足够机动能力调整轨道高度, 设置最大推力比如下:

$$T_p = 0.2\frac{\mu_E}{R_E^2}, \qquad T_e = 0.1\frac{\mu_E}{R_E^2}.$$

式(20)中的光滑因子ρ和ϵ设置如下:

$$\rho = 100, \qquad \epsilon = 0.01. \tag{28}$$

通过循环DE求解框架给出式(25)中的优化参数值及对应的J的计算结果如下:

$$\begin{cases} \boldsymbol{\lambda}_p = [0.80, -1.78, 1.86, 0.29, 0.010, -0.73]^{\mathrm{T}}, \\ \boldsymbol{\lambda}_e = [-3.51, -8.66, -1.56, -1.49, -5.41, -7.11]^{\mathrm{T}}, \\ t_f = 3.70, \quad J = 198.74. \end{cases}$$

为展示循环DE求解框架的求解效率和精度, 将其与经典DE算法在用时和优化值两方便进行了对比. 其中目标函数取用文献[29]中的评价指标$h(p)$, 其意义为$\boldsymbol{C}_{eq}$的各分量绝对值的加权和, 对比结果如表1所示.

图3展示了追踪航天器和逃逸航天器的运动轨迹. 为揭示双方在博弈中的策略, 图4和5展示了双方控制方向的坐标分量和在局部坐标系中的方向角θ和ψ. 从图4可以看出, 双方的控制方向坐标分量具有相似的变化趋势, 这是由于双方的初始状态相近, 因此控制轨迹

表 1 DE算法性能对比

Table 1 Comparison of performances with DE algorithm

算法	时间消耗 (s)	目标函数$h(p)$
循环DE求解框架	1384.41	3.37
经典DE	11540	407

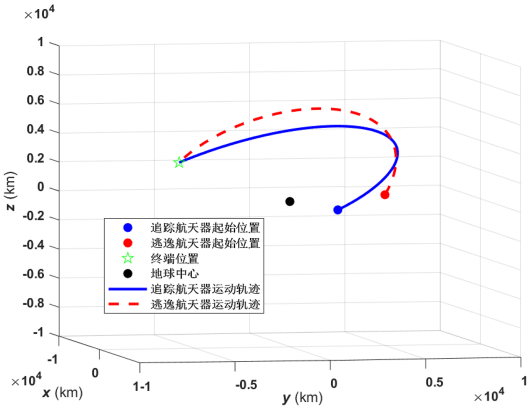

图 3 (网络版彩图)运动轨迹

Figure 3 (Color online) Motion trajectories.

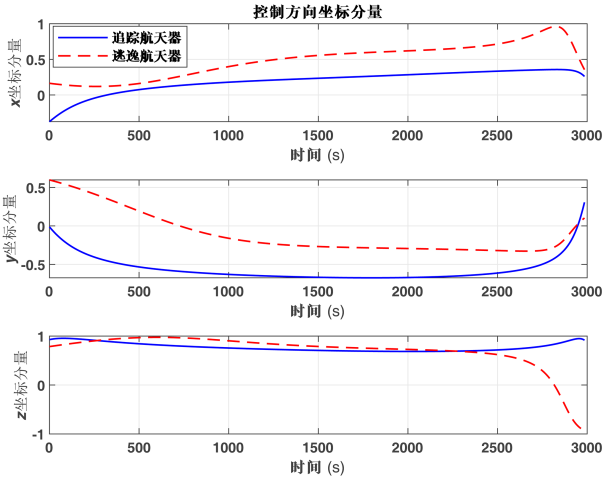

图 4 (网络版彩图)控制方向坐标分量

Figure 4 (Color online) Control direction components.

在全局坐标系中体现出追踪效应. 控制方向角则一定程度反映了双方经纬度和轨道高度的变化, 其中θ体现了航天器在局部坐标系的控制方向, ψ则揭示了轨道高度的变化趋势, 图5则反映了如下特征: 双方航天器呈现向局部东北方向施加推力, 追踪航天器轨道高度趋向于先降低后升高, 逃逸航天器轨道高度则整体上持续升高.

图6展示了追踪航天器和逃逸航天器的轨道高度变化轨迹. 如前文所述, 追踪航天器轨道高度先降低后升高, 而逃逸航天器轨道高度持续升高. 图7展示了双

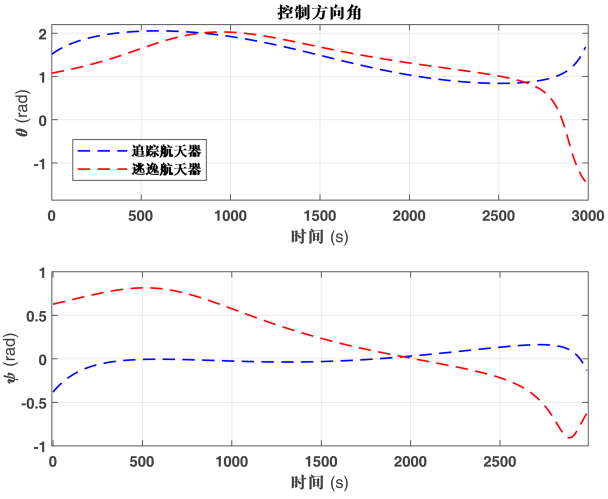

图 5 (网络版彩图)控制方向角

Figure 5 (Color online) Control direction angles.

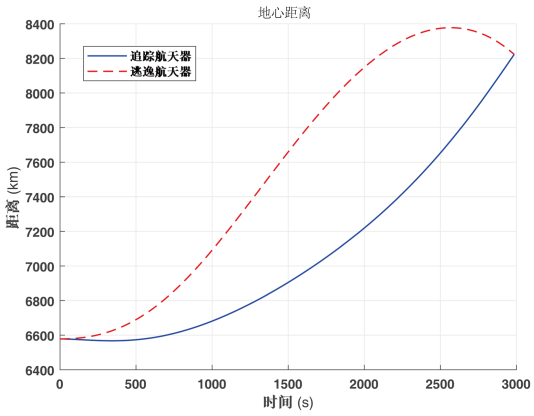

图 6 (网络版彩图)地心距离轨迹

Figure 6 (Color online) Geographic distance trajectories.

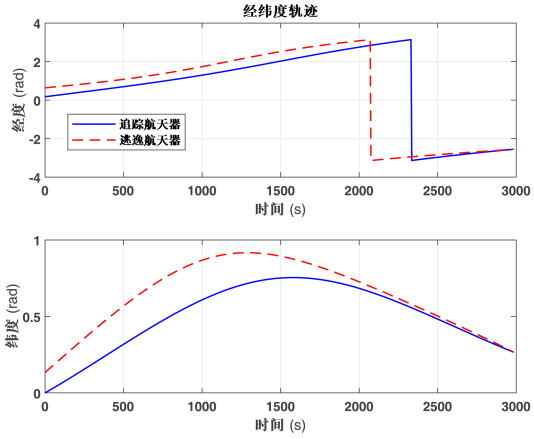

图 7 (网络版彩图)经纬度变化轨迹

Figure 7 (Color online) Latitude and longitude trajectories.

方航天器的经纬度变化轨迹. 由于双方初始位置相近, 因此整体经纬度变化趋势相同, 值得注意的一点是, 因为轨道约束的存在, 双方需要经过约半个轨道周期完成交会, 即纬度轨迹体现成一个拱形状. 因此, 追踪航天器为了加速交会时间, 选择降低轨道高度以加速纬度变化速度, 继而升高轨道高度以完成交会. 而逃逸航天器, 本应采取同样的策略以增加完成交会时所经历的轨道位移, 但由于轨道高度约束的存在, 其策略更变为持续提高轨道高度, 通过提高追踪航天器升高轨道高度的距离来延长博弈持续的时间.

为验证上述分析, 本文后续对未添加轨道约束的轨道追逃博弈问题进行了仿真模拟, 并与前述仿真结果进行了对比.

图8对比了添加轨道约束和未添加轨道约束工况下双方的运动轨迹, 图9则对比了两种工况下双方的轨道高度轨迹. 两张图均体现出, 添加轨道高度后, 显著增高了博弈过程和博弈终端双方的轨道高度. 在图9中, 未添加轨道高度约束工况下, 追踪航天器和逃逸航天器均先降低轨道高度后提升轨道高度, 这意味着, 添加轨道约束将会限制双方策略的执行. 此外, 值得注意的一点是, 未添加轨道高度约束工况下, 双方航天器的地心距离均达到地球半径R_E以下, 这代表双方航天器运动到了地球内部, 这显然是不合理的.

综上所述, 轨道约束的添加规避了航天器与地球碰撞或进入地球内部等不合理的行为, 同时会限制策略的执行, 致使博弈终端交会的轨道高度增加, 间接延长博弈时间: 从2095 s延长至2985 s.

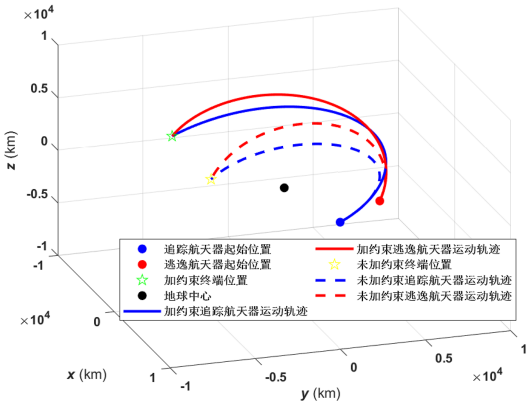

图 8 (网络版彩图)运动轨迹对比

Figure 8 (Color online) Comparison of motion trajectories.

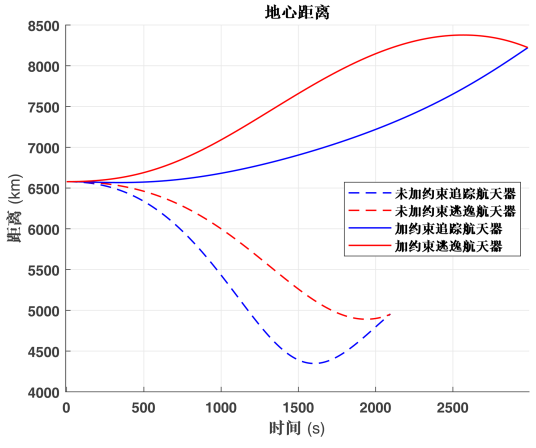

图 9 (网络版彩图)地心距离轨迹对比

Figure 9 (Color online) Comparison of geographic distance trajectories.

4.3 循环DE求解框架性能对比验证

为验证本文提出的循环DE求解框架的性能, 本节选用了文献[29]中的另外3个工况, 并通过仿真模拟, 验证了循环DE求解框架的求解效率及鲁棒性.

本节选用的工况分别代表高度差异轨道、顺行逆行轨道、椭圆轨道, 其初始工况设置如下.

Case 1 不同轨道高度

$$P\Rightarrow\begin{cases}x_{p0}=-6598.210\,\text{km},\\ y_{p0}=1163.440\,\text{km},\\ z_{p0}=0\,\text{km},\\ v_{xp0}=-0.458\,\text{km/s},\\ v_{yp0}=-2.598\,\text{km/s},\\ v_{zp0}=7.248\,\text{km/s},\end{cases}\quad E\Rightarrow\begin{cases}x_{p0}=-6755\,\text{km},\\ y_{p0}=-3900\,\text{km},\\ z_{p0}=0\,\text{km},\\ v_{xp0}=3.520\,\text{km/s},\\ v_{yp0}=-6.097\,\text{km/s},\\ v_{zp0}=1.241\,\text{km/s}.\end{cases}$$

Case 2 顺逆行轨道

$$P\Rightarrow\begin{cases}x_{p0}=-6598.210\,\text{km},\\ y_{p0}=1163.440\,\text{km},\\ z_{p0}=0\,\text{km},\\ v_{xp0}=1.160\,\text{km/s},\\ v_{yp0}=6.578\,\text{km/s},\\ v_{zp0}=3.857\,\text{km/s},\end{cases}\quad E\Rightarrow\begin{cases}x_{p0}=6062.180\,\text{km},\\ y_{p0}=3500\,\text{km},\\ z_{p0}=0\,\text{km},\\ v_{xp0}=-3.268\,\text{km/s},\\ v_{yp0}=5.660\,\text{km/s},\\ v_{zp0}=3.773\,\text{km/s}.\end{cases}$$

Case 3 椭圆轨道

$$P\Rightarrow\begin{cases}x_{p0}=536.291\,\text{km},\\ y_{p0}=-6008.420\,\text{km},\\ z_{p0}=-3470.030\,\text{km},\\ v_{xp0}=7.402\,\text{km/s},\\ v_{yp0}=1.063\,\text{km/s},\\ v_{zp0}=-0.138\,\text{km/s},\end{cases}\quad E\Rightarrow\begin{cases}x_{p0}=-5824.010\,\text{km},\\ y_{p0}=1571.490\,\text{km},\\ z_{p0}=-3470.030\,\text{km},\\ v_{xp0}=2.332\,\text{km/s},\\ v_{yp0}=7.105\,\text{km/s},\\ v_{zp0}=-0.138\,\text{km/s}.\end{cases}$$

为保持与文献[29]一致, 设置追逃双方最大推力比及轨道高度约束如下:

$$T_p=0.1\frac{\mu_E}{R_E^2},\qquad T_e=0.05\frac{\mu_E}{R_E^2},$$

$$\text{Low}_i=R_E,\qquad \text{Upper}_i=+\infty.$$

通过循环DE求解框架给出式(25)中的优化参数值及对应的J的计算结果如下:

Case 1

$$\begin{cases}\boldsymbol{\lambda}_p = [-3.57, -2.03, 1.46, -4.79, 1.03, -0.38]^{\mathrm{T}}, \\ \boldsymbol{\lambda}_e = [3.55, 2.67, -0.90, -4.23, 8.24, 0.14]^{\mathrm{T}}, \\ t_f = 3.42, \quad J = 90.69, \end{cases}$$

Case 2

$$\begin{cases}\boldsymbol{\lambda}_p = [0.73, -0.67, -0.70, -0.26, -1.62, -1.24]^{\mathrm{T}}, \\ \boldsymbol{\lambda}_e = [-1.63, -1.89, -0.66, -0.59, -1.64, -1.79]^{\mathrm{T}}, \\ t_f = 1.39, \quad J = 73.31, \end{cases}$$

Case 3

$$\begin{cases}\boldsymbol{\lambda}_p = [0.91, -1.30, -1.23, -0.41, -2.31, -0.64]^{\mathrm{T}}, \\ \boldsymbol{\lambda}_e = [4.31, -3.57, 2.48, 0.69, -7.61, 1.11]^{\mathrm{T}}, \\ t_f = 2.40, \quad J = 102.46. \end{cases}$$

图10–12展示了Cases 1–3下, 追逃双方的轨道高度变化轨迹.

表2展示了Cases 1–3下, 循环DE求解框架与文献[29]中算法在求解时间和求解精度上的对比, 其中每个工况计算10次并取平均用时和平均误差.

5 结论

本文研究了一种包含轨道高度约束的轨道追逃博弈问题. 根据庞特里亚金极大值原理, 推导了该问题的最优性条件, 并基于方程系数矩阵的性质进行了降维.基于罚函数方法对过程约束进行处理, 并将轨道追逃博弈问题转化为最优化问题. 针对该最优化问题, 提出了一种循环DE求解框架以兼顾全局搜索能力和收敛效率. 仿真结果表明:

(1) 循环DE求解框架显著提高DE求解算法的求解效率, 并能够大幅提高问题的求解精度.

(2) 添加轨道高度约束能够避免航天器与地球的体积碰撞, 具备现实意义. 同时, 轨道高度约束会对双方博弈策略的执行产生影响, 减小其策略的执行程度以达成不违反轨道高度约束的目标.

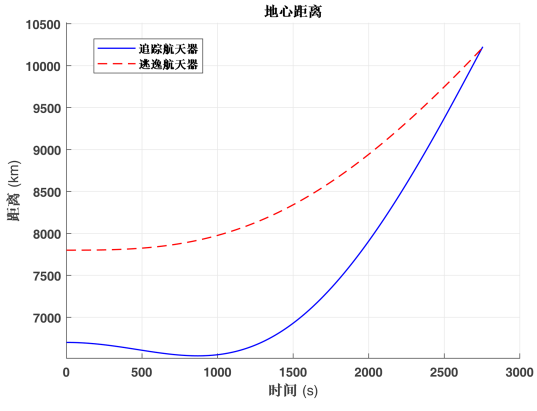

图 10 (网络版彩图) Case 1地心距离轨迹

Figure 10 (Color online) Geographic distance trajectories for Case 1.

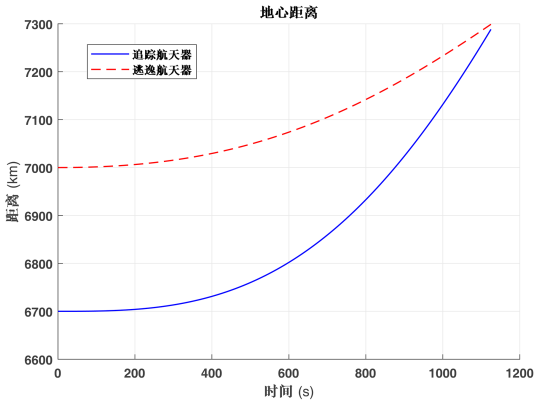

图 11 (网络版彩图) Case 2地心距离轨迹

Figure 11 (Color online) Geographic distance trajectories for Case 2.

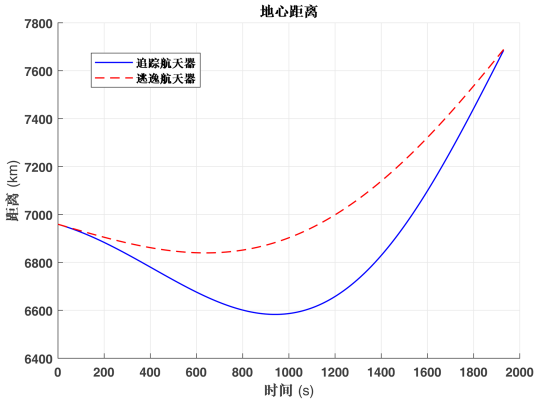

图 12 (网络版彩图) Case 3地心距离轨迹

Figure 12 (Color online) Geographic distance trajectories for Case 3.

表 2 Cases 1–3 算法性能对比

Table 2 Comparison of algorithm performances for Cases 1–3

	Case 1		Case 2		Case 3	
	时间 (s)	$h(p)$	时间 (s)	$h(p)$	时间 (s)	$h(p)$
本文	56.76	7.19	56.76	2.35	133.92	2.97
文献[29]	70.11	260.16	90.15	170.65	261.01	90.15

参考文献

1 Isaacs R. Differential Games. New York: Wiley, 1965. 278–280
2 Breakwell J V, Merz A W. Minimum required capture radius in a coplanar model of the aerial combat problem. AIAA J, 1977, 15: 1089–1094
3 Guelman M, Shinar J, Green A. Qualitative study of a planar pursuit evasion game in the atmosphere. J Guid Control Dyn, 1990, 13: 1136–1142
4 Macias V, Becerra I, Murrieta-Cid R, et al. Image feedback based optimal control and the value of information in a differential game. Automatica, 2018, 90: 271–285
5 Isaacs R. Games of Pursuit. Santa Monica: Rand Corporation, 1951
6 Bryson A E, Ho Y C, Siouris G M. Applied optimal control: Optimization, estimation, and control. IEEE Trans Syst Man Cybern, 1979, 9: 366–367
7 Başar T, Olsder G J. Dynamic Noncooperative Game Theory. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics, 1998
8 Zhang Q H, Sun Y, Huang M M, et al. Pursuit-evasion barrier of two spacecrafts under minute continuous radial thrust in coplanar orbit. Control Decis, 2007, 22: 530–534
9 Raivio T, Ehtamo H. Visual aircraft identification as a pursuit-evasion game. J Guid Control Dyn, 2000, 23: 701–708
10 Hargraves C R, Paris S W. Direct trajectory optimization using nonlinear programming and collocation. J Guid Control Dyn, 1987, 10: 338–342
11 Breitner M H, Pesch H J, Grimm W. Complex differential games of pursuit-evasion type with state constraints, Part 1: Necessary conditions for optimal open-loop strategies. J Opt Theor Appl, 1993, 78: 419–441
12 Breitner M H, Pesch H J, Grimm W. Complex differential games of pursuit-evasion type with state constraints, Part 2: Numerical computation of optimal open-loop strategies. J Opt Theor Appl, 1993, 78: 443–463
13 Hafer W T, Reed H L, Turner J D, et al. Sensitivity methods applied to orbital pursuit evasion. J Guid Control Dyn, 2015, 38: 1118–1126
14 Pontani M, Conway B A. Numerical solution of the three-dimensional orbital pursuit-evasion game. J Guid Control Dyn, 2009, 32: 474–487
15 Li Z, Zhu H, Yang Z, et al. A dimension-reduction solution of free-time differential games for spacecraft pursuit-evasion. Acta Astronaut, 2019, 163: 201–210
16 Li Z, Zhu H, Yang Z, et al. Saddle point of orbital pursuit-evasion game under J_2-perturbed dynamics. J Guid Control Dyn, 2020, 43: 1733–1739
17 Zhang Z, Zhang Y, Wang B. Application of the hp-adaptive pseudospectral method in spacecraft orbit pursuit-evasion game. Adv Space Res, 2024, 73: 1597–1610
18 Pang B, Wen C, Han H, et al. Solving pursuit/evasion game along elliptical orbit by providing precise gradient. J Guid Control Dyn, 2024, 47: 797–807
19 Jiang R, Ye D, Xiao Y, et al. Orbital interception pursuit strategy for random evasion using deep reinforcement learning. Space Sci Technol, 2023, 3: 0086
20 Tian Y, Liao H, Xu J, et al. Unsupervised spectrum anomaly detection method for unauthorized bands. Space Sci Technol, 2022, 2022: 9865016
21 Li M J, Li M, Liu Y F, et al. A review on the development of spaceborne membrane antennas. Space Sci Technol, 2022, 2022: 9803603
22 Li J F, Zeng X Y. Flight dynamics in the gravitational fields of irregular asteroids (in Chinese). Adv Mech, 2017, 47: 429–451 [李俊峰, 曾祥远. 不规则小行星引力场内的飞行动力学. 力学进展, 2017, 47: 429–451]
23 Zeng X, Liu X. Searching for time optimal periodic orbits near irregularly shaped asteroids by using an indirect method. IEEE Trans Aerosp Electron Syst, 2017, 53: 1221–1229
24 Han H, Dang Z. Models and strategies for J_2-perturbed orbital pursuit-evasion games. Space Sci Technol, 2023, 3: 0063
25 Shi M, Ye D, Sun Z, et al. Spacecraft orbital pursuit-evasion games with J_2 perturbations and direction-constrained thrust. Acta Astronaut, 2023, 202: 139–150
26 Chambers M L. The mathematical theory of optimal processes. J Oper Res Soc, 1965, 16: 493–494
27 Bai F S. Exact Penalty Methods in Nonlinear Programming (in Chinese). Dissertation for Doctoral Degree. Shanghai: Shanghai University, 2003 [白富生. 非线性规划中的精确罚函数. 博士学位论文. 上海: 上海大学, 2003]
28 Sallam K M, Elsayed S M, Chakrabortty R K, et al. Improved multi-operator differential evolution algorithm for solving unconstrained problems. In: Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC). Glasgow: IEEE, 2020. 1–8
29 Zeng X, Yang L, Zhu Y, et al. Comparison of two optimal guidance methods for the long-distance orbital pursuit-evasion game. IEEE Trans Aerosp Electron Syst, 2021, 57: 521–539

A cyclic differential evolution framework for constrained orbital pursuit-evasion games

ZHANG QuFei[1,2], JI MingJiang[2], YAN Bin[2], LIU CaiShan[1*] & CAO Lu[2*]

[1]*College of Engineering, Peking University, Beijing 100871, China*
[2]*National Innovation Institute of Defense Technology, Academy of Military Science, Beijing 100071, China*
**Corresponding authors (LIU CaiShan, email: LIUCS@pku.edu.cn; CAO Lu, email: caolu_space2015@163.com)*

Orbital pursuit-evasion games have been widely studied, but most of the research is based on the assumption of no process constraints. For orbital pursuit-evasion games subject to orbital height constraints, the conditions for the existence of saddle points are derived. Based on the properties of the coefficient matrix, the equations are decoupled and reduced in dimension to two two-point boundary value problems (TPBVPs). The penalty function method is used to handle process constraints, and the two TPBVPs are then transformed into an optimization problem for the optimal conditions. This paper proposes a cyclic differential evolution framework for solving this optimization problem and conducts simulation. The simulation results verify the efficiency and accuracy of the framework, and reveal the practical significance of orbital height constraints for preventing collisions between spacecraft and the Earth.

orbital pursuit-evasion games, orbital height constraints, penalty function method, two-point boundary value problem, cyclic differential evolution framework

PACS: 02.60.Cb, 02.50.Le, 95.10.Eg, 02.30.Yy

doi: 10.1360/SSPMA-2024-0310