

基于计算博弈的脉冲作用下航天器追逃策略

顾天妍¹, 张永合², 蒋 峻², 李超勇¹

(1. 浙江大学 电气工程学院, 浙江 杭州 310027; 2. 中科院上海微小卫星工程中心, 上海 201203)

摘 要: 本文研究了地球高阶引力模型中, 基于脉冲作用的航天器轨道追逃问题。针对该问题, 将航天器轨道追逃问题定义为两选手计算博弈, 其中以博弈双方的距离和燃料消耗设计了性能指标函数, 并以速度增量大小和方向构建容许控制集。此外, 为了保证纳什均衡解的求解效率, 引入了快速搜索(ARS)算法, 并设计了一种数据剪枝方法用于优化搜索空间, 最终实现了基于计算博弈的控制策略的快速求解。仿真结果表明: 该方法能够有效解决脉冲作用下航天器的追逃难题, 与传统方法相比, 该方法能同时满足脱靶量精度和燃料消耗要求, 具有一定的可行性和有效性。

关键词: 航天器追逃; 博弈; 脉冲; 非合作目标; 纳什均衡

中图分类号: V 412.4

文献标志码: A

DOI: 10.19328/j.cnki.2096-8655.2023.04.011

Spacecraft Pursuit-Evasion Strategy Based on Computational Game with Impulse Maneuvers

GU Tianyan¹, ZHANG Yonghe², JIANG Jun², LI Chaoyong¹

(1. College of Electrical Engineering, Zhejiang University, Hangzhou 310027, Zhejiang, China;

2. Shanghai Engineering Center for Microsatellites, Chinese Academy of Sciences, Shanghai 201203, China)

Abstract: The problem of spacecraft pursuit-evasion with impulse maneuvers in the high-order gravity model of the Earth is investigated. The spacecraft pursuit-evasion problem is formulated as a two-player game, where the performance index functions are designed based on the distance and fuel consumption between the players, and an admissible control set is constructed based on the magnitude and direction of the impulse maneuvers. Moreover, to ensure the efficiency of finding Nash equilibrium solutions, the action-reaction search (ARS) algorithm is introduced, and a data pruning method is designed to optimize the search space. Ultimately, fast solving is achieved with the control strategy based on the computational game. The simulation results demonstrate that the proposed approach can effectively solve the challenging problem of spacecraft pursuit-evasion with impulse maneuvers. Especially, compared with traditional methods, this method can simultaneously meet the requirements of miss distance accuracy and fuel consumption, and has certain feasibility and validity.

Key words: spacecraft pursuit-evasion; game; impulse maneuver; non-cooperative target; nash equilibrium

0 引言

近年来, 随着航天器控制技术和轨道规划技术的不断发展, 航天器空间交会技术逐渐成熟, 同时能兼容的目标形式也逐渐增加^[1]。当航天器双方都有自主机动能力时, 传统的单边最优控制策略不再

适用, 航天器的交会问题可视为双边控制问题, 即航天器追逃博弈问题。相较于经典控制策略, 航天器追逃博弈控制同时考虑了双方的控制信息, 更适用于非合作机动目标, 受到国内外学者的广泛关注。

收稿日期: 2023-03-21; 修回日期: 2023-04-10

基金项目: 国家自然科学基金(62088101), 浙江省杰出青年基金(LR20F030003)

作者简介: 顾天妍(1998—), 女, 硕士, 主要研究方向为航天器追逃博弈。

通信作者: 张永合(1977—), 男, 博士, 研究员, 主要研究方向为导航制导与控制。

针对航天器追逃博弈问题,目前大多数学者都采用了微分对策的方法进行研究。ISAACS^[2]提出微分对策论,将控制论中的部分概念和原理与博弈论结合,并应用于二人追逃问题中,初步形成了微分对策论。在此基础上,STUPIK等^[3]利用Clohessy-Wiltshire(CW)方程将航天器追逃博弈问题转化为非线性两点边值问题,并通过克里金法求解了航天器的追逃策略。针对微分对策理论方程求解困难的问题,吴其昌等^[4]分别采用了遗传算法、差分进化算法和蚁群算法来求解牛顿迭代初值,从而避免复杂微分方程的求解,但这类方法的计算时间较长。此外,PONTANI等^[5]利用半直接配点方法求解异面情况下航天器追逃问题,并通过仿真验证了该方法的鲁棒性。

随着人工智能和大数据处理技术的发展,利用人工智能方法进行在线决策和规划成为可能,近年来在航天器追逃问题中的应用也逐渐受到重视。许旭升等^[6]提出了一种基于多智能体深度强化学习的集群卫星空间轨道追逃博弈方法,通过多智能体深度决定性策略梯度法(Multi-agent Deep Deterministic Policy Gradient, MADDPG)训练数据,最终得到各卫星的策略。刘冰雁等^[7]在传统强化学习的基础上,建立了模糊推理模型,利用分支深度强化学习有效解决了行为数量与映射规则的组合增长问题,缩短了仿真时间,提高了仿真效率。吴其昌等^[8]将深度神经网络应用到航天器追逃博弈中,搭建了4层神经网络,并通过Adam优化算法对网络进行了训练,同时验证了最终生成策略与真实策略相近,实现了在线决策。

值得注意的是,上述文献都是针对航天器在连续时间下的连续推力模型展开,且目前大部分研究工作均在连续机动的基础上进行,针对脉冲作用下的轨道追逃博弈的研究较少,同时模型的相关研究和结论并不成熟。针对这一问题,VENIGALLA等^[9-10]提出了可达集的概念,并证明了逃逸航天器在知道追踪航天器的可达集的前提下能成功完成逃逸,给出了在共面时逃逸航天器的最优逃逸方向。LIU等^[11]研究了三人博弈问题,仅在一次脉冲机动的前提下,利用粒子群算法和牛顿插值法解决了异面最优轨迹求解问题。于大腾^[12]基于序列二次优化算法建立了追踪器多脉冲最优交会模型,采用遗传算法进行了机动优化,提升了飞行器的空间生存能力。

上述方法都有效解决了航天器轨道追逃问题,但其中航天器的动力学模型由简化的CW方程进行描述,多数没有考虑摄动力因素的影响^[13]。同时,非圆轨道和较大的相对距离也是CW方程误差的主要来源。然而航天器的实际轨迹会受到各种不可避免的摄动影响,特别是当航天器在低轨道和中轨道运行时,摄动作用力会对线性模型下的博弈结果产生不可忽略的负面影响。因此,本文旨在解决地球高阶引力模型下,基于脉冲控制的航天器轨道追逃问题,并通过计算博弈进行求解。

在计算博弈问题中,快速搜索(Action-Reaction Search, ARS)算法能够高效求解纳什均衡点。针对多组动态武器目标分配(Multi-team Dynamic Weapon Target Assignment, MDWTA)生成矩阵规模较大的问题,GALATI^[14]提出利用ARS算法沿最优方向搜索,有效利用内存,提高计算可行性。剪枝算法常用于减少算法搜索时间和提高计算效率,REED^[15]提出利用剪枝算法应用于神经网络中以减少不必要的搜索路径,至今剪枝已经分化出多种类别,包括结构化剪枝和非结构化剪枝等,应用于计算机^[16]、电气^[17]、航空航天^[18]等领域。

本文提出了一种基于计算博弈的控制策略,通过优化航天器的速度增量大小和方向,实现了追逃双方的性能指标最优化。本文的主要工作:1)完成了航天器轨道追逃博弈问题的数学定义,其中性能指标函数考虑博弈双方的距离和燃料消耗,并以速度增量大小和方向构建容许控制集;2)创新性地引入ARS算法,并将一种数据剪枝策略嵌入,从而保证了纳什均衡解的准确性以及求解速度。本文证明了在逃逸航天器没有机动的情况下,该算法可以成功地将轨道追逃博弈问题转换为最基本的航天器轨道交会问题。仿真结果验证了本文方法的有效性和可行性。

1 问题描述与数学建模

假设在航天器轨道追逃任务场景中,追击航天器预先通过Hohmann转移拦截目标航天器,但在通过第一次机动之后,目标航天器可通过施加一个微小偏移脉冲规避追击航天器的拦截。此时,双方各自有一次施加脉冲机动的机会,且在同一时刻机动。追击航天器需要在燃料消耗尽可能小的情况下拦截逃逸航天器,而逃逸航天器则需在燃料消耗

尽可能小的情况下规避拦截。

1.1 航天器动力学建模

在航天器追逃博弈问题中,为了便于计算和分析,采用 J2000 下的地球惯性坐标系。考虑到摄动力等因素,航天器的动力学模型为^[19]

$$\begin{aligned}\dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= -\frac{\partial U}{\partial \mathbf{r}}\end{aligned}\quad (1)$$

式中: $\mathbf{r}$ 为航天器的位置矢量; $\mathbf{v}$ 为航天器的速度矢量; U 为地球的引力势函数。

当地球为旋转椭球体,且只考虑 J_1 至 J_6 摄动项时,令地球赤道半径为 R_e ,则地球引力势函数可以简化为

$$U = \frac{\mu}{r} (1 - J_2 P_2 R_e^2 - J_3 P_3 R_e^3 - J_4 P_4 R_e^4 - J_5 P_5 R_e^5 - J_6 P_6 R_e^6) \quad (2)$$

式中: μ 为地球引力常数; $J_2=1.082\ 6\times 10^{-3}$, $J_3=-2.536\times 10^{-6}$, $J_4=-1.618\ 6\times 10^{-6}$, $J_5=-0.226\times 10^{-6}$, $J_6=0.539\times 10^{-6}$; $P_2\sim P_6$ 为勒让德多项式,表达式如下:

$$\begin{cases} P_2 = 1.5 \left(\frac{z}{r} \right)^2 - 0.5 \\ P_3 = 2.5 \left(\frac{z}{r} \right)^3 - 1.5 \left(\frac{z}{r} \right) \\ P_4 = \left[35 \left(\frac{z}{r} \right)^4 - 30 \left(\frac{z}{r} \right)^2 + 3 \right] / 8 \\ P_5 = \left[63 \left(\frac{z}{r} \right)^5 - 70 \left(\frac{z}{r} \right)^3 + 1.5 \left(\frac{z}{r} \right) \right] / 8 \\ P_6 = \left[231 \left(\frac{z}{r} \right)^6 - 315 \left(\frac{z}{r} \right)^4 + 105 \left(\frac{z}{r} \right)^2 - 5 \right] / 16 \end{cases} \quad (3)$$

在施加脉冲作用的时刻,航天器的状态变化为

$$\begin{cases} \mathbf{r}^+ = \mathbf{r}^- = \mathbf{r} \\ \mathbf{v}^+ = \mathbf{v}^- + \Delta \mathbf{v} \end{cases} \quad (4)$$

式中:上标“-”和“+”分别为脉冲作用前后的状态。

在 J2000 坐标系下,式(4)可以扩展为

$$\begin{cases} \mathbf{r}^+ = \mathbf{r}^- = \mathbf{r} \\ v_x^+ = v_x^- + \Delta v \cos \varphi \cos \gamma \\ v_y^+ = v_y^- + \Delta v \sin \varphi \cos \gamma \\ v_z^+ = v_z^- + \Delta v \sin \gamma \end{cases} \quad (5)$$

式中: φ 为脉冲推力偏角; γ 为脉冲推力仰角。

1.2 博弈问题构建

自 20 世纪以来,航天器的交会对接技术在工程应用及理论研究中具有重要意义,有许多突出成果,其中 Hohmann 转移^[20]和 Lambert 追击^[21]是最为经典的方法。Hohmann 转移方法给出了共面下轨道转移的最小能量消耗,奠定了之后大多数理论的基础,但是存在调相时间过长,耗费时间巨大的问题;Lambert 追击方法计算了固定时间下两点之间轨道转移所需脉冲。上述理论均要求目标航天器被动飞行且没有自主机动,不适用于目标存在自主机动的航天器追逃博弈问题。在航天器追逃博弈问题中,追击航天器和逃逸航天器通过控制自身的脉冲机动,使得双方的性能指标函数达到最优,即:

寻找 (u_P^*, u_E^*)

$$\text{使得} \begin{cases} \max J_P(u_P^*, u_E^*) \\ \max J_E(u_P^*, u_E^*) \end{cases} \quad (6)$$

式中:下标 P、E 分别为追击航天器和逃逸航天器。

航天器的追逃博弈问题包含 3 个要素:博弈参与者 $\{P, E\}$;双方各自的性能指标函数 J ;以及参与者的行为策略 (u_P, u_E) 。

追击航天器的行为策略 u_P 定义为

$$u_P = [\Delta v, \varphi, \gamma] \quad (7)$$

上式中各项满足如下约束条件:

$$\begin{cases} 0 \leq \Delta v \leq \Delta v_{\max} \\ \varphi_{\min} \leq \varphi \leq \varphi_{\max} \\ \gamma_{\min} \leq \gamma \leq \gamma_{\max} \end{cases} \quad (8)$$

针对上述航天器追逃博弈问题,SCHEERES 等^[9]基于可达集的概念推导了逃逸航天器的最优逃逸方向,耿远卓等^[22]利用终端诱导强化学习对航天器追逃博弈问题进行了求解,通过在奖励函数中考虑终端误差从而提高追击成功率。然而,这些航天器通常采用二体模型或 CW 方程进行求解,没有考虑地球摄动因素影响,且对轨道形状有所限制,所得结果精度不足。为了满足实际情况,提升求解模型的精度,本文采用计算博弈的方法来解决航天器追逃博弈问题。

与传统的以解析形式求解博弈双方纳什均衡点的方法不同,计算博弈通过对双方的策略进行数值搜索,得到双方各自的最优解。此外,追逃航天器的性能指标函数 J 包含两部分,分别由追逃双方的距离以及各自消耗的燃料定义。双方博弈的目的是通过给出自己的脉冲控制策略,使得相应的性

能指标函数最大化。具体来说,对于追击方而言,其期望在尽量减少燃料消耗的情况下减少追逃双方距离;对于逃逸方而言,则期望自己在尽量减少燃料消耗的情况下增加追逃双方距离。因此,性能指标 J_P 和 J_E 定义如下:

$$\begin{aligned} J_P &= -w_P^1 L - w_P^2 \Delta v_P \\ J_E &= w_E^1 L - w_E^2 \Delta v_E \end{aligned} \quad (9)$$

式中: w_P^1 、 w_P^2 、 w_E^1 、 w_E^2 分别为权重系数; L 为追逃双方距离; Δv_P 为追击航天器消耗的燃料; Δv_E 为逃逸航天器消耗的燃料。

权重系数应满足以下条件:

$$\begin{cases} w_P^1 + w_P^2 = 1 \\ w_E^1 + w_E^2 = 1 \\ 0 < w_P^1, w_P^2, w_E^1, w_E^2 < 1 \end{cases} \quad (10)$$

追逃航天器双方的距离 L 定义如下:

$$L = \sqrt{(x_P - x_E)^2 + (y_P - y_E)^2 + (z_P - z_E)^2} \quad (11)$$

当逃逸航天器中途没有脉冲机动,则追逃问题就会转化为普通的轨道交会问题。此时,双方的性能指标函数也相应变化。对于追击方而言,性能指标如下:

$$J_P = -L \quad (12)$$

对于逃逸方而言,性能指标 J_E 如下:

$$J_E = L \quad (13)$$

$$\begin{bmatrix} J_P(u_P^1, u_E^1), J_E(u_P^1, u_E^1) & \cdots & J_P(u_P^1, u_E^{n-1}), J_E(u_P^1, u_E^n) \\ J_P(u_P^2, u_E^1), J_E(u_P^2, u_E^1) & \cdots & J_P(u_P^2, u_E^{n-1}), J_E(u_P^2, u_E^n) \\ \vdots & \vdots & \vdots \\ J_P(u_P^m, u_E^1), J_E(u_P^m, u_E^1) & \cdots & J_P(u_P^m, u_E^{n-1}), J_E(u_P^m, u_E^n) \end{bmatrix} \quad (16)$$

通过计算式(16)中的矩阵解可以求得纳什均衡点。决策矩阵式(16)存在维数大、计算时间长和搜索效率低的问题,为了解决这一问题,本文采用ARS算法来求解博弈矩阵。

2 基于计算博弈的快速优化策略

为了求解追逃双方的纳什均衡点,使得双方的性能指标函数处于最优,本文采用ARS算法^[24]来进行快速求解,具体算法流程如下。

2.1 算法1: Action-Reaction Search

输入: 最大迭代次数 M ;

输出: 纳什均衡点 (u_P^*, u_E^*) ;

步骤1 初始化迭代次数 $i = 1$;

步骤2 初始化策略 $(\tilde{u}_P, \tilde{u}_E)$;

本文所定义的追逃博弈问题就是寻找追击航天器者和逃逸航天器的纳什均衡点 (u_P^*, u_E^*) 问题,使其指标函数大于任意其余策略的指标函数,纳什均衡点的定义如下:

$$\begin{aligned} J_P(u_P^*, u_E^*) &\geq J_P(u_P, u_E^*), \forall u_P \in U_P \\ J_E(u_P^*, u_E^*) &\geq J_E(u_P^*, u_E), \forall u_E \in U_E \end{aligned} \quad (14)$$

式中: U_P 、 U_E 分别为追击航天器和逃逸航天器的所有策略。

对于追逃航天器双方而言,可供选择的策略是有限的,由纳什均衡的存在性定理^[23]可知,每一个有限的策略式博弈至少存在一个由式(14)描述的纳什均衡解。那么最终求得的纳什均衡解所对应的速度增量大小和方向就是当前问题的一个可行解。

在基于脉冲控制下的航天器追逃博弈中,追击航天器最优策略 u_P^* 和逃逸航天器最优策略 u_E^* 存在如下形式:

$$\begin{aligned} u_P^* &= [\Delta v_P^*, \varphi_P^*, \gamma_P^*] \\ u_E^* &= [\Delta v_E^*, \varphi_E^*, \gamma_E^*] \end{aligned} \quad (15)$$

由此,追击航天器策略 $u_P^i (i = 1, 2, \dots, m)$ 和逃逸航天器策略 $u_E^i (i = 1, 2, \dots, n)$ 共同构成了博弈的决策矩阵,用计算博弈可以求得相应的 $m \times n$ 维决策矩阵为

步骤3 寻找 $\hat{u}_P$ 使得 $J_P(\hat{u}_P, \tilde{u}_E) \geq J_P(u_P, \tilde{u}_E)$,

$\forall u_P \in U_P$;

步骤4 令 $\tilde{u}_P = \hat{u}_P$;

步骤5 寻找 $\hat{u}_E$ 使得 $J_E(\tilde{u}_P, \hat{u}_E) \geq J_E(\tilde{u}_P, u_E)$,

$\forall u_E \in U_E$;

步骤6 if $(\hat{u}_E == \tilde{u}_E)$ then;

步骤7 $u_P^* = \hat{u}_P, u_E^* = \hat{u}_E$;

步骤8 算法结束;

步骤9 else;

步骤10 if $(i > M)$ then;

步骤11 未找到纳什均衡;

步骤12 算法结束;

步骤13 else;

步骤14 $\tilde{u}_E = \hat{u}_E$;

步骤15 $i = i + 1$;

步骤 16 执行步骤 3;

步骤 17 end。

由于在步骤 4 中 $\tilde{u}_p = \hat{u}_p$ 和步骤 6 中 $\hat{u}_E = \tilde{u}_E$, 如果算法 1 能够进入到步骤 7, 那么步骤 3 和步骤 5 中的式子与式(14)中的纳什点定义一致, $(\hat{u}_p, \hat{u}_E)$ 就是所求问题的纳什均衡点 (u_p^*, u_E^*) 。

2.2 算法 2: 数据剪枝方法

与穷举法相比, ARS 算法无需对整个矩阵进行遍历, 能够有效地减少计算维度, 具有计算效率高、时间快的优点。对于航天器追逃博弈这一问题而言, 实时性和快速性尤为关键, 也为 ARS 算法的使用提供了有力依据。

博弈决策矩阵式(16)中会存在不满足实际约束的策略对, ARS 算法同样也对这些策略进行了搜索, 增加了计算时间, 使搜索效率降低。因此, 本文提出了一种剪枝方法, 在矩阵生成和搜索过程中, 对无需计算的元素进行标记剪枝, 剪枝流程如下:

输入: 追逃博弈双方的策略集合 $U_p = [\Delta v_p, \varphi_p, \gamma_p]$, $U_E = [\Delta v_E, \varphi_E, \gamma_E]$, 收益维度 M, N ;

输出: 无需计算的矩阵元素标记集合 A^{pruned} ;

步骤 1 初始化索引标记 $m = 1, n = 1, p = 0, q = 0$;

步骤 2 while($m \leq M$) and ($n \leq N$) do;

步骤 3 if($\Delta v_p^m == 0$);

步骤 4 $p = p + 1$;

步骤 5 if($p \neq 1$);

步骤 6 更新标记集合: $(m, n) \rightarrow A^{\text{pruned}}$;

步骤 7 if($\Delta v_E^n == 0$);

步骤 8 $q = q + 1$;

步骤 9 if($q \neq 1$);

步骤 10 更新标记集合: $(m, n) \rightarrow A^{\text{pruned}}$;

步骤 11 if($n = N$) then;

步骤 12 $m = m + 1$;

步骤 13 $n = 1$;

步骤 14 else;

步骤 15 $n = n + 1$;

步骤 16 返回 A^{pruned} 。

算法 2 通过对冗余不必要的数据进行剪枝, 减少了不必要的搜索过程, 提高了搜索速度和运算效率。

本文通过使用 ARS 算法准确寻找纳什均衡点,

ARS 算法通过对单独的行列寻找来避免搜索整个矩阵, 极大地提高了运行效率。本文采用剪枝技巧来提升寻找纳什均衡点的计算速度, 通过对冗余数据的剪枝达到缩小矩阵规模的目的, 从而加快数值搜索进度。

为了更直接地说明基于剪枝的快速搜索方法的优越性, 将其与穷举法、 α - β 剪枝^[25]进行对比。在进行不同维度的矩阵计算时, 双方求解纳什均衡点的时间如图 1 所示。

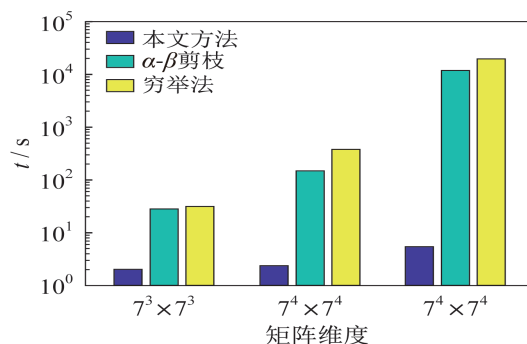


图 1 决策时间对比

Fig. 1 Comparison of decision time

由图 1 可知, 相比于其他方法, 本文所采用的快速求解算法能有效提高计算效率, 节省计算时间。当矩阵维度扩大时, 其余方法的仿真时间以指数形式增长, 而本文方法增幅不大, 仍能快速寻找到纳什均衡解。

3 仿真分析

3.1 轨道交会算例分析

当逃逸航天器不施加机动时, 追逃问题演变为轨道交会问题。本文采用硬件平台为 3.60 GHz AMD Ryzen 5 3500X 处理器, 软件平台为 Matlab R2022b 进行仿真。为了便于比较, 将追击航天器的初始机动时间与 Hohmann 转移的时间一致。两航天器初始轨道六根数见表 1。

表 1 交会场景下航天器的初始六根数

Tab. 1 Initial six parameters of the spacecrafts in the rendezvous scenario

航天器	a/km	e	$i/(^{\circ})$	$\omega/(^{\circ})$	$\Omega/(^{\circ})$	$f/(^{\circ})$
P	100 10	0.2	30	0	10	88
E	100 90	0.2	30	0	10	90

两航天器飞行轨迹如图 2 所示。

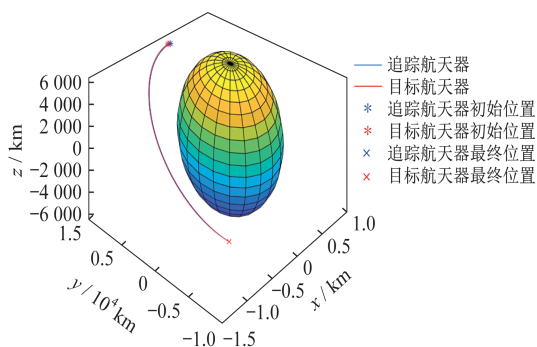


图 2 交会场景下航天器三维空间飞行轨迹

Fig. 2 Trajectories of the spacecraft in the rendezvous scenario

为进一步验证本文方法的有效性,分别与 Hohmann 转移和 Lambert 追击进行对比,三者都在同一时刻进行第一次脉冲机动,且都施加两次机动,一致采用地球高阶引力模型。追击航天器与逃逸航天器相对距离如图 3 所示。

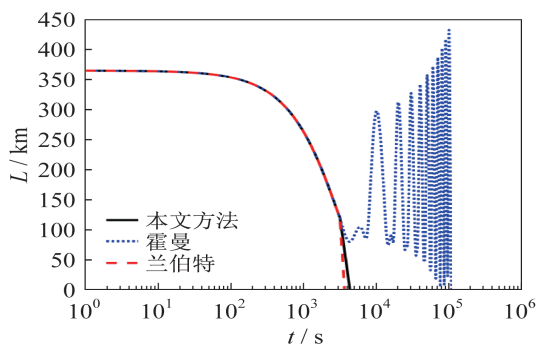


图 3 交会场景下航天器相对距离

Fig. 3 Relative distance of the spacecraft in the rendezvous scenario

3 种方法的最终计算结果见表 2。其中,追击航天器均在 $T_1=3\,172.590\,5$ s 时施加第 1 次脉冲机动,第 1 次速度增量大小为 Δv_1 ,第 2 次机动时刻为 T_2 ,第 2 次速度增量大小为 Δv_2 。

由表 2 可知,与 Hohmann 转移相比,本文方法的脱靶量明显较小,但稍逊于 Lambert 追击。本文方法在保证脱靶量的情况下,使用速度增量明显小于 Lambert 追击,有效减少了燃料消耗。此外,在进行计算时,本文方法无需选定初值,避免了采用其他 2 种方法在初值选定不正确时可能存在的不收敛和脱靶量大的问题。

表 2 交会场景下航天器仿真结果

Tab. 2 Simulation results of the spacecraft in the rendezvous scenario

参数	本文方法	Hohmann 转移	Lambert 追击
T_1/s	3 172.590 5	3 172.5905	3 172.590 5
$\Delta v_1/(\text{km}\cdot\text{s}^{-1})$	-0.072 0	0.031 1	0.202 2
T_2/s	4 293.731 0	103 980.221 7	3 422.589 9
$\Delta v_2/(\text{km}\cdot\text{s}^{-1})$	-0.048 0	-0.018 8	0.000 2
$\Delta v/(\text{km}\cdot\text{s}^{-1})$	0.120 0	0.050 0	0.202 4
T/s	4 349.181 0	109 017.919 5	3 672.657 9
脱靶量/km	0.004 3	0.978 5	0.000 3

3.2 追逃算例分析

当逃逸航天器施加机动时,两者是典型的追逃问题。在追击航天器采用 Hohmann 转移施加第一次机动后,双方都还有一次脉冲机动机会,逃逸航天器施加脉冲推力逃离追击航天器,双方在这一时刻通过矩阵搜索都使彼此性能指标函数达到最优。

根据 Hohmann 转移计算易得,追击航天器第一次机动时刻 $T_1=3\,172.590\,5$ s,施加速度增量大小 $\Delta v_1=0.031\,1$ km/s。此后,逃逸航天器施加控制量试图远离追击航天器,双方展开追逃博弈。设置追击航天器最大可使用速度增量 $\Delta v_p^{\max}=0.061\,0$ km/s,逃逸航天器最大可使用速度增量 $\Delta v_e^{\max}=0.008\,0$ km/s。两航天器飞行轨迹如图 4 所示。

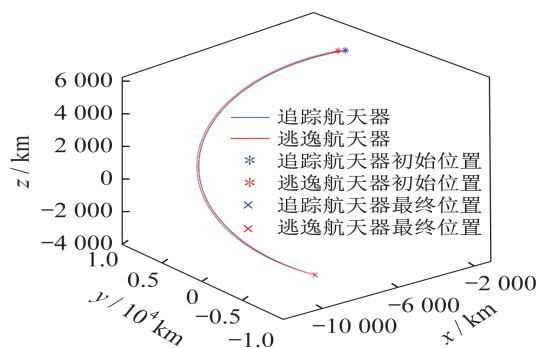


图 4 追逃航天器三维空间飞行轨迹

Fig. 4 Three-dimensional space flight trajectory of the spacecraft in the pursuit-evasion game

两航天器使用燃料情况和脱靶量见表 3。通过仿真可知,追击航天器使用了全部速度增量 $\Delta v_p=0.061\,0$ km/s,偏角 $\varphi_p=199.998\,4^\circ$,仰角 $\gamma_p=-5.998\,4^\circ$ 。逃逸航天器也使用了全部速度增量 $\Delta v_e=0.008\,0$ km/s,偏角 $\varphi_e=270^\circ$,仰角

$\gamma_E = -5.9984^\circ$ 。两航天器距离从最开始的 364.95 km 缩短到最终脱靶量 4.32 km。当采用这种策略时,双方的性能指标函数都达到最优。

表 3 航天器追逃博弈的仿真结果

Tab. 3 Simulation results of the spacecraft in the pursuit-evasion game

T_2/s	$\Delta v_P/(km \cdot s^{-1})$	$\Delta v_E/(km \cdot s^{-1})$	脱靶量/km	T/s
3 172.610 5	0.061 0	0.008 0	4.315 8	4 396.060 5

4 结束语

本文提出了一种基于计算博弈的航天器追逃博弈策略,采用了 ARS 算法求解纳什均衡点,同时利用剪枝策略缩小了决策矩阵维度,减少了不必要的搜索过程,提高了搜索效率。通过实验仿真,得到如下结论:

1) 针对基于脉冲的航天器追逃问题,本文提出的方法能够较好地求得双方最优策略,且求解时间迅速。相对于传统方法,本文模型精度较高,考虑了地球摄动等非线性因素,且对轨道形状和初始距离没有限制,与实际偏差较小,可行性高,具有较强的鲁棒性。

2) 针对基于脉冲的轨道交会问题,相对于 Lambert 追击和 Hohmann 转移,本文提出的方法在保证脱靶量精度较高的情况下,消耗燃料较少,无需考虑初值问题。

参考文献

- [1] 程国采. 航天飞行器最优控制理论与方法[M]. 北京: 国防工业出版社, 1999: 285-287.
- [2] ISSACS R. Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization [M]. New York: John Wiley & Sons, 1965: 334.
- [3] STUPIK J, PONTANI M, CONWAY B. Optimal pursuit/evasion spacecraft trajectories in the hill reference frame [C]// AIAA/AAS Astrodynamics Specialist Conference. Minneapolis, Minnesota: American Institute of Aeronautics and Astronautics, 2012.
- [4] 吴其昌, 张洪波. 基于生存型微分对策的航天器追逃策略及数值求解[J]. 控制与信息技术, 2019, (4): 39-43.
- [5] PONTANI M, CONWAY B A. Numerical solution of the three-dimensional orbital pursuit-evasion game [J].

Journal of Guidance, Control, and Dynamics, 2009, 32 (2): 474-487.

- [6] 许旭升, 党朝辉, 宋斌, 等. 基于多智能体强化学习的轨道追逃博弈方法[J]. 上海航天(中英文), 2022, 39(2): 24-31.
- [7] 刘冰雁, 叶雄兵, 高勇, 等. 基于分支深度强化学习的非合作目标追逃博弈策略求解[J]. 航空学报, 2020, 41 (10): 343-353.
- [8] 吴其昌, 李彬, 李君, 等. 基于深度神经网络的无限时域型航天器追逃策略求解[J]. 航天控制, 2019, 37(6): 13-18, 58.
- [9] VENIGALLA C, SCHEERES D. Delta-v-based analysis of spacecraft pursuit - evasion games [J]. Journal of Guidance, Control, and Dynamics, 2021, 44 (11): 1961-1971.
- [10] VENIGALLA C, SCHEERES D. Spacecraft rendezvous and pursuit/evasion analysis using reachable sets [C]//2018 Space Flight Mechanics Meeting. Kissimmee, Florida: American Institute of Aeronautics and Astronautics, 2018.
- [11] LIU Y, LI R, HU L, et al. Optimal solution to orbital three-player defense problems using impulsive transfer [J]. Soft Computing, 2018(22): 2921-2934.
- [12] 于大腾. 空间飞行器安全防护规避机动方法研究[D]. 长沙: 国防科技大学, 2017.
- [13] 周敬, 郑建华, 李明涛. J2 摄动下卫星相对运动解析方程研究[J]. 航天控制, 2017(2): 44-50.
- [14] GALATI D G. Game theoretic target assignment strategies in competitive multi-team systems [D]. Pennsylvania: University of Pittsburgh, 2005.
- [15] REED R. Pruning algorithms-a survey [J]. IEEE Transactions on Neural Networks, 1993, 4 (5): 740-747.
- [16] HUANG G B, SARATCHANDRAN P, SUNDARARAJAN N. A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation [J]. IEEE Transactions on Neural Networks, 2005, 16(1): 57-67.
- [17] 张翼, 朱永利. 基于深度-广度联合剪枝的电力设备局部放电轻量化诊断方法[J]. 电工技术学报, 2023, 38 (7): 1935-1945, 1955.
- [18] 蒋昭阳. 基于深度学习的非合作航天器单目视觉位姿测量方法[D]. 哈尔滨: 哈尔滨工业大学, 2020.
- [19] 荆武兴, 高长生. 空间飞行器导航、制导与控制[M]. 哈尔滨: 哈尔滨工业大学出版社, 2018: 69.
- [20] GOBETZ F W, DOLL J R. A survey of impulsive Trajectories[J]. AIAA Journal, 1969, 7(5): 801-834.

- [21] BATTIN R H. An Introduction to the Mathematics and Methods of Astrodynamics[M]. New York: American Institute of Aeronautics and Astronautics, 1987: 1-32.
- [22] 耿远卓,袁利,黄煌,等. 基于终端诱导强化学习的航天器轨道追逃博弈[J]. 自动化学报, 2023, 49(5): 974-984.
- [23] 罗云峰. 博弈论教程[M]. 北京: 清华大学出版社, 2007: 59-61.
- [24] ZHANG T, LI C, MA D, et al. An optimal task management and control scheme for military operations with dynamic game strategy[J]. Aerospace Science and Technology, 2021, 115: 106815.
- [25] KNUTH D E, MOORE R W. An analysis of alpha-beta pruning[J]. Artificial Intelligence, 1975, 6(4): 293-326.

(上接第 53 页)

- [19] MIAO Y, DENG H, LIU X, et al. multi-authority attribute-based keyword search over encrypted cloud data[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(4): 1667-1680.
- [20] YANG K, SHU J, XIE R. Efficient and provably secure data selective sharing and acquisition in cloud-based systems[J]. IEEE Transactions on Information Forensics and Security, 2022, 18: 71-84.
- [21] YIN H, LI Y F, DENG H, et al. Practical and dynamic attribute-based keyword search supporting numeric comparisons over encrypted cloud data[J]. IEEE Transactions on Services Computing, 2022: 1-14.
- [22] LI Y, LI J. Lightweight public key authenticated encryption with keyword search against adaptively-chosen-targets adversaries for mobile devices[J]. IEEE Transactions on Mobile Computing, 2022, 21(12): 4397-4409.
- [23] DAN B, LYNN B, SHACHAM H. Short signatures from the Weil pairing[J]. Journal of Cryptology, 2004, 17(4): 297-319.
- [24] MIAO Y B, WENG J, LIU X M, et al. Enabling verifiable multiple keywords search over encrypted cloud data[J]. Information Sciences, 2018, 465: 21-37.
- [25] PENG Y G, CUI J C, PENG C G, et al. Certificateless public key Encryption with keyword search[J]. China Communications, 2014, 11(11): 100-113.