

航天器交会型轨道追逃策略的滚动时域优化

张乘铭, 朱彦伟*, 杨乐平, 杨傅云翔

(国防科技大学空天科学学院, 湖南长沙 410073)

摘要:针对航天器自由时域交会型轨道追逃过程中的测量误差等不确定性对交会的影响,提出了一种基于滚动时域优化的高时效策略求解方法。根据微分对策理论推导得到博弈鞍点控制策略,并对问题进行等价转换;通过预先离线求解开环鞍点策略,将问题初始状态和相应的解作为样本以进行神经网络训练,训练后的网络结构可以快速得到相应问题的近似解。为了更好地应对博弈环境中的测量噪声,基于神经网络结构设计了滚动时域求解框架,并通过周期性的滚动求解最终实现对逃逸航天器的交会。数值仿真表明,所提出的策略可以有效应对测量噪声不确定性,且相比于文献中已有的策略,计算耗时可从分钟级降至秒级。

关键词:追逃博弈;微分对策;深度神经网络;滚动时域优化

中图分类号:V448.2 文献标志码:A 开放科学(资源服务)标识码(OSID):

文章编号:1001-2486(2024)03-021-09



听语音
与作者互动
聊科研

Receding horizon optimization for spacecraft pursuit-evasion strategy in rendezvous

ZHANG Chengming, ZHU Yanwei*, YANG Leping, YANG Fuyunxiang

(College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China)

Abstract: Given the influence of uncertainty such as measurement errors in the process of spacecraft free-time orbital pursuit-evasion game for rendezvous, a high-efficiency strategy based on receding horizon optimization was proposed as a solution method. The saddle point control strategy of the game was derived according to differential games, and the equivalent transformation of the problem was carried out. By solving open-loop saddle point strategy off-line in advance, the initial states of the problem and the corresponding solutions were taken as samples for neural network training, and the trained network structure can quickly obtain the approximate solution of the corresponding problem. In order to better deal with the measurement noise in the game environment, a receding horizon optimization framework was designed based on the neural network structure. By periodically solving the problem, the rendezvous of the pursuer and evader was finally realized. Numerical simulation shows that the proposed strategy can effectively deal with the uncertainty of measurement noise, and compared with the existing strategy in the literature, the calculation time can be reduced from minutes to several seconds.

Keywords: pursuit-evasion game; differential games; deep neural network; receding horizon optimization

近年来在空间态势感知和在轨操控领域,多项技术试验与验证项目^[1]相继展开,包括地球同步轨道空间态势感知计划、太空监视小卫星系统、局部空间自主导航与制导试验卫星以及 EAGLE 卫星和 Mycroft 卫星等^[2],航天器近距离操控任务逐步成为研究热点。其中,航天器追逃博弈为其中一种典型的任务场景。在此问题中具有冲突目标的双方航天器基于自身感知能力和机动能力通过施加控制开展博弈对抗,直至满足博弈终止条件。该问题中,通常以微分方程的形式表示双方

航天器的动力学过程,以微分对策理论^[3]为基础建立零和博弈模型,并通过数值算法获得双方均衡策略。

当前围绕航天器追逃问题,大多将拦截作为博弈终止条件,即要求在终端时刻双方位置重合。Pontani 等^[4]将遗传算法与非线性规划方法结合提出了一种半直接配点法对远距离的拦截博弈问题进行了研究,降低了问题维度并避免了直接求解边值问题。Carr 等^[5]结合比例导引法提出了一种协态变量初值猜测方法,并将基于微分对策

理论推导得到的两点边值问题转化为单边问题进行数值求解。Shen 等^[6]同样研究了这一问题,首先考虑了逃逸方无机动力时的单边问题,并将这一问题的解作为初始解,进一步得到追逃问题的解。Zhang 等^[7]关注了共面轨道的追逃问题,并分别针对开始于捕获区和逃逸区内的问题设计了神经网络算法和强化学习算法进行了近似求解。Shi 等^[8]对含 J2 摄动力影响且推力角度受限的拦截型博弈问题进行了研究,将逃逸方无机动力情况下的解作为初始猜测。

上述工作中并未对博弈终止时的速度进行约束,从而导致此种接近方式存在较大碰撞风险。为保证在博弈终端时刻安全接近,本文关注交会型追逃问题并开展了研究工作。关于此类问题,常燕等^[9]运用微分对策对共面机动目标交会问题进行研究,并采用了非线性规划算法得到了追逃双方的最优轨迹。Tartaglia 等^[10]基于状态相关系数参数化和状态相关黎卡提方程方法求解了非线性动力学条件下航天器交会过程中非零和博弈的纳什均衡解。Prince 等^[11]研究了以时间为优化目标的椭圆轨道交会问题,并应用遗传算法进行了数值求解,但计算耗时较长,且所得开环策略难以应对复杂博弈环境。Venigalla 等^[12]通过对脉冲次数和施加脉冲位置设置了较强约束条件,以经典轨道六要素视角对脉冲推力方式的交会型问题进行了研究,使得航天器的无动力飞行阶段得以高效利用。从以上研究工作中可以发现,由于增加了速度约束,问题维度和求解难度进一步增加,且所设计的求解算法难以应对含有测量误差的复杂博弈环境。

为解决以上问题,论文针对近距离航天器交会型轨道追逃问题进行了研究,博弈终止条件由双方的相对位置和相对速度共同决定。为体现问题的强对抗性,不考虑追逃过程的燃料消耗而将终端时间作为博弈目标,即追踪方希望以尽可能短的时间实现交会,而逃逸方则尽可能延迟这一时刻。在后续内容中,基于零和微分对策理论对自由时域的追逃问题进行鞍点策略的推导并进行了问题的等价转换;设计了神经网络结构通过生成样本并训练从而快速生成近似轨迹;基于上述神经网络进一步设计了滚动时域优化框架用于求解含有测量噪声的追逃问题,并进行了仿真实验和分析。

1 问题建模

为便于问题描述,在不影响一般性的前提下,

后续研究工作将博弈场景设置为地球同步轨道 (geosynchronous orbit, GEO) 附近。

1.1 近距离交会相对运动模型

图 1 给出了博弈问题的初始场景,当两航天器距离较近且二者之间的相对距离远小于 GEO 高度时,选择相对运动模型描述博弈过程更为方便。以 GEO 轨道作为参考轨道,在该轨道上选择位于两航天器附近的一虚拟动点作为坐标系原点,进一步引入局部垂直/局部水平 (local vertical local horizontal, LVLH) 坐标系对追逃航天器的相对运动进行描述。

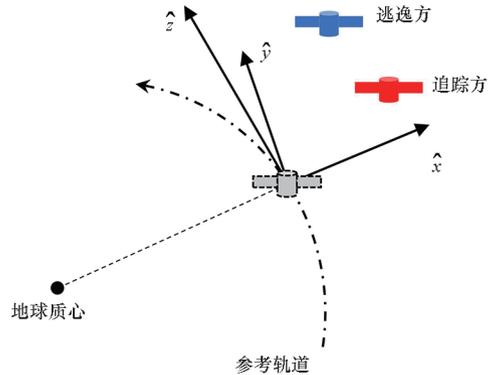


图 1 问题场景

Fig. 1 Scenario of the problem

如图 1 所示,在该坐标系中, \hat{x} 轴方向沿地球质心指向该虚拟动点, \hat{z} 轴指向参考轨道面法向, \hat{y} 轴则沿速度方向并构成右手系。进而在二体运动假设下,可以采用希尔方程或 CW (Clohessy-Wiltshire) 方程描述该坐标系下的相对运动,如式(1)所示,其中: \mathbf{A} 和 \mathbf{B} 为系数矩阵, \mathbf{x} 表示航天器的状态,由位置 \mathbf{r} 和速度 \mathbf{v} 组成,忽略博弈过程中航天器的质量变化, T 表示航天器提供的加速度大小且为常数^[13], \mathbf{u} 表示提供加速度的方向矢量。下标 i 则用于区分博弈双方,规定 $i = p$ 为追踪方, $i = e$ 为逃逸方。图 2 和式(2)给出了加速度方向矢量与推力角度的关系。

$$\dot{\mathbf{x}}_i = \mathbf{A}\mathbf{x}_i + T_i\mathbf{B}\mathbf{u}_i \quad (1)$$

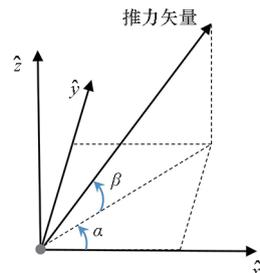


图 2 推力矢量方向示意图

Fig. 2 Thrust vector direction diagram

$$\mathbf{u} = \begin{bmatrix} \cos\alpha\cos\beta \\ \sin\alpha\cos\beta \\ \sin\beta \end{bmatrix}$$

$$\alpha \in [-\pi, \pi], \beta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \quad (2)$$

1.2 交会微分博弈模型

追逃双方从零时刻开始施加控制量展开博弈对抗,其初始状态分别记为 \mathbf{x}_{p0} 和 \mathbf{x}_{e0} 。当追逃双方位置和速度满足交会条件时,当前博弈过程结束,则终端约束条件如式(3)所示,其中, t 表示时间,下标 f 表示博弈结束。

$$\Psi(\mathbf{x}_{pf}, \mathbf{x}_{ef}, t_f) = \begin{bmatrix} \mathbf{r}_{pf} - \mathbf{r}_{ef} \\ \mathbf{v}_{pf} - \mathbf{v}_{ef} \end{bmatrix} = \mathbf{0}_{6 \times 1} \quad (3)$$

二人零和博弈目标函数则为:

$$J = t_f \quad (4)$$

因此,求解该微分博弈问题即找到鞍点处的控制策略 \mathbf{u}_p^* 和 \mathbf{u}_e^* 使得:

$$J(\mathbf{u}_p^*, \mathbf{u}_e) \leq J(\mathbf{u}_p^*, \mathbf{u}_e^*) \leq J(\mathbf{u}_p, \mathbf{u}_e^*) \quad (5)$$

式(5)的含义为:在鞍点处双方策略互为最优,且任意一方执行偏离鞍点解的控制策略都将使其处于更为不利的处境^[14]。

根据微分对策理论,为求解上述鞍点策略,首先引入 Hamilton 函数:

$$H = \boldsymbol{\lambda}_p^T \dot{\mathbf{x}}_p + \boldsymbol{\lambda}_e^T \dot{\mathbf{x}}_e \quad (6)$$

式中, $\boldsymbol{\lambda}$ 为协态变量,与航天器状态量相关。

$$\boldsymbol{\lambda}_p = [\lambda_{p1}, \lambda_{p2}, \lambda_{p3}, \lambda_{p4}, \lambda_{p5}, \lambda_{p6}]^T \quad (7)$$

$$\boldsymbol{\lambda}_e = [\lambda_{e1}, \lambda_{e2}, \lambda_{e3}, \lambda_{e4}, \lambda_{e5}, \lambda_{e6}]^T \quad (8)$$

定义广义终端函数为:

$$\Phi = t_f + \mathbf{v}^T \Psi \quad (9)$$

式中,

$$\mathbf{v} = [v_1, v_2, v_3, v_4, v_5, v_6]^T \quad (10)$$

\mathbf{v} 是与终端约束条件相关的 Lagrange 乘子。

进一步可知协态变量需满足:

$$\begin{cases} \dot{\boldsymbol{\lambda}}_i = -\frac{\partial H}{\partial \mathbf{x}_i} \\ \boldsymbol{\lambda}_{if} = \frac{\partial \Phi}{\partial \mathbf{x}_{if}} \end{cases} \quad (11)$$

考虑到该问题终端时间 t_f 不固定,还需要引入终端时刻横截条件:

$$\frac{\partial \Phi}{\partial t_f} + H_f = 1 + H_f = 0 \quad (12)$$

此外鞍点条件下追逃双方的控制策略需满足:

$$\begin{cases} \mathbf{u}_p^* = \arg \min_{\mathbf{u}_p} H \\ \mathbf{u}_e^* = \arg \max_{\mathbf{u}_e} H \end{cases} \quad (13)$$

式(13)可以展开得到:

$$\begin{cases} \frac{\partial H}{\partial \mathbf{u}_p^*} = 0 \\ \frac{\partial^2 H}{\partial \mathbf{u}_p^{*2}} \geq 0 \\ \frac{\partial H}{\partial \mathbf{u}_e^*} = 0 \\ \frac{\partial^2 H}{\partial \mathbf{u}_e^{*2}} \leq 0 \end{cases} \quad (14)$$

计算式(14),可以得到鞍点策略中的推力方向为:

$$\begin{cases} \cos\alpha_i^* = \kappa \frac{\lambda_{i4}}{\sqrt{\lambda_{i4}^2 + \lambda_{i5}^2}} \\ \sin\alpha_i^* = \kappa \frac{\lambda_{i5}}{\sqrt{\lambda_{i4}^2 + \lambda_{i5}^2}} \\ \sin\beta_i^* = \kappa \frac{\lambda_{i6}}{\sqrt{\lambda_{i4}^2 + \lambda_{i5}^2 + \lambda_{i6}^2}} \end{cases} \quad (15)$$

其中,

$$\kappa = \begin{cases} -1 & i = p \\ 1 & i = e \end{cases}$$

根据式(15),可以发现鞍点控制策略仅和相应航天器的协态变量有关,在此基础上通过进一步展开式(11)得到:

$$\dot{\boldsymbol{\lambda}}_i = -\mathbf{A}^T \boldsymbol{\lambda}_i \quad (16)$$

$$\boldsymbol{\lambda}_{pf} = -\boldsymbol{\lambda}_{ef} = \mathbf{v} \quad (17)$$

式(17)表示,当双方均采用鞍点对策时,在 t_f 时刻追逃双方协态变量未知但互为相反数,结合式(16),可以得到:

$$\boldsymbol{\lambda}_p(t) = -\boldsymbol{\lambda}_e(t) \quad (18)$$

将式(18)代入式(15),则有:

$$\mathbf{u}^* = \mathbf{u}_p^* = \mathbf{u}_e^* \quad (19)$$

至此,式(1)、式(3)、式(9)~(13)组成了一个两点边值问题,其中追逃双方的初始状态由问题给出,控制策略由 \mathbf{v} 和 t_f 确定,当满足交会条件时终止。

1.3 微分博弈问题的等价转换

根据1.2节中的交会微分对策模型,可以知道当问题初始条件确定时(假设该初始条件存在鞍点解),可以通过求解 \mathbf{v} 和 t_f 获得鞍点策略。在完全理性假设条件下,将所得的 \mathbf{u}_p^* 和 \mathbf{u}_e^* 代入航天器动力学方程,即:

$$\dot{\mathbf{x}}_p = \mathbf{A}\mathbf{x}_p + T_p \mathbf{B}\mathbf{u}^* \quad (20)$$

$$\dot{\mathbf{x}}_e = \mathbf{A}\mathbf{x}_e + T_e \mathbf{B}\mathbf{u}^* \quad (21)$$

两式作差可得:

$$\dot{\mathbf{x}}_p - \dot{\mathbf{x}}_e = \mathbf{A}(\mathbf{x}_p - \mathbf{x}_e) + (T_p - T_e)\mathbf{B}\mathbf{u}^* \quad (22)$$

定义 $\mathbf{x}_{pe} = \mathbf{x}_p - \mathbf{x}_e$, $T_{pe} = T_p - T_e$, 则有:

$$\dot{\mathbf{x}}_{pe} = \mathbf{A}\mathbf{x}_{pe} + T_{pe}\mathbf{B}\mathbf{u}^* \quad (23)$$

式(23)给出了当追逃双方执行鞍点策略时其相对状态变化的动力学方程,表明该鞍点策略 \mathbf{u}^* 与初始状态为 \mathbf{x}_{pe} 、加速度大小为 T_{pe} 、终止条件为 $\mathbf{x}_{pe}(t_f) = \mathbf{0}_{6 \times 1}$ 、优化目标为 t_f 的最优控制问题中的最优策略是相同的。

在前期研究中,Prince 等^[11]应用 MATLAB 中的遗传算法(genetic algorithm, GA)工具箱求解 \mathbf{v} 和 t_f 进而得到 \mathbf{u}^* ,但耗时较长,且得到的开环解难以应对追逃过程中的不确定性。但可以通过求解最优控制问题获得原追逃微分对策问题的解。基于这一思想,进一步将深度神经网络与 Radau 伪谱法结合,并采用滚动时域优化框架对含有噪声误差的交会博弈问题进行求解。

2 基于深度神经网络生成近似解

根据 1.2 节和 1.3 节的分析发现, \mathbf{u}^* 可由满足鞍点必要条件的 \mathbf{v} 和 t_f 确定,这一过程通常需要通过数值优化算法多次迭代求解。但由于 \mathbf{v} 并无实际物理意义,难以生成较为精确的初始猜测值以减少计算耗时。由于近年来大数据和深度神经网络在动力学与控制领域的广泛应用^[15-17],可根据应用场景预先生成大量追逃样本用于训练神经网络,从而拟合问题初始状态和待求未知量之间的关系,为数值优化算法提供初始猜测,从而提高算法效率。

2.1 协态变量预处理

通过式(15)可以发现, $\mathbf{u}^*(t)$ 与矢量 $[\lambda_{i4}(t), \lambda_{i5}(t), \lambda_{i6}(t)]^T$ 的模长无关,而仅与其在空间的指向有关。再考虑 λ_i 的动力学方程式(16)~(17)可知, \mathbf{u}^* 仅由 $\mathbf{v}/\|\mathbf{v}\|$ 和 t_f 确定。这一结论表明在通过数值优化算法(如遗传算法)对追逃问题求解过程中,针对同一初始条件,由于优化算法的随机性,可能会得到不同的 \mathbf{v} ,从而导致问题初始状态与获得的解之间对应关系更为复杂。因此为提高深度神经网络训练精度,需要对其进行归一化处理。

因为 \mathbf{u}^* 仅由 $\mathbf{v}/\|\mathbf{v}\|$ 和 t_f 确定,令

$$\mathbf{v}_0 = \frac{\mathbf{v}}{\|\mathbf{v}\|} \quad (24)$$

则 \mathbf{v}_0 为 6 维空间中的单位球,可以通过 5 个变量对表示以减少计算维度。

进一步考虑到两航天器位置变量和速度变量之间量级差距较大,令

$$\mathbf{v}_{p0} = [\mathbf{v}_{01}, \mathbf{v}_{02}, \mathbf{v}_{03}]^T = \|\mathbf{v}_{p0}\| \begin{bmatrix} \cos\alpha_1 \cos\beta_1 \\ \sin\alpha_1 \cos\beta_1 \\ \sin\beta_1 \end{bmatrix} \quad (25)$$

$$\mathbf{v}_{ve} = [\mathbf{v}_{04}, \mathbf{v}_{05}, \mathbf{v}_{06}]^T = \|\mathbf{v}_{ve}\| \begin{bmatrix} \cos\alpha_2 \cos\beta_2 \\ \sin\alpha_2 \cos\beta_2 \\ \sin\beta_2 \end{bmatrix} \quad (26)$$

式(25)和(26)分别表示在终端时刻,与位置和速度相关的协态变量组成的向量,进一步定义:

$$\varepsilon = \frac{\|\mathbf{v}_{p0}\|}{\|\mathbf{v}_{ve}\|} \quad (27)$$

因此, \mathbf{v}_0 可由 $[\varepsilon, \alpha_1, \beta_1, \alpha_2, \beta_2]$ 表示,即 \mathbf{v} 通过上述过程可由 $[\varepsilon, \alpha_1, \beta_1, \alpha_2, \beta_2]$ 表示。通过这一归一化过程避免了网络训练中不必要的输出,并一定程度减轻了由于 \mathbf{v}_{p0} 和 \mathbf{v}_{ve} 量级差别较大带来的网络训练难度,后续在 2.3.1 节中对神经网络样本数据预处理过程进行了进一步说明。

2.2 追逃问题样本生成

根据 1.3 节和 2.1 节的分析,每一个样本中的输入为 $\boldsymbol{\eta}$ 、输出为 $\boldsymbol{\sigma}$,其中:

$$\boldsymbol{\eta} = [\mathbf{x}_{pe}^T(t_0), T_{pe}] \quad (28)$$

$$\boldsymbol{\sigma} = [\varepsilon, \alpha_1, \beta_1, \alpha_2, \beta_2, t_f] \quad (29)$$

由于论文主要关注 GEO 附近的近距离航天器交会型追逃问题,因此在样本生成过程中选择 $a = 42\ 378.136\ 6\ \text{km}$ 的圆轨道作为参考轨道,参考轨道倾角为 0° 。表 1 给出了在生成样本过程中初始时刻 \mathbf{x}_{pe} 和 T_{pe} 的范围 Ω 。其中, LB 和 UB 分别表示下限和上限, $x_{pe1}, x_{pe2}, x_{pe3}$ 表示位置矢量的三个分量, $x_{pe4}, x_{pe5}, x_{pe6}$ 则表示三个速度分量。 $g = \mu_E/R_E^2$, 其中 μ_E 为地球引力常数, R_E 为地球平均半径。

表 1 样本生成范围 Ω

Tab. 1 Range of generated samples Ω

样本范围	$x_{pe1}, x_{pe2}, x_{pe3}/\text{km}$	$x_{pe4}, x_{pe5}, x_{pe6}/(\text{m/s})$	$T_{pe}(g)$
LB	-100	-10	10^{-4}
UB	100	10	10^{-3}

算法 1 给出了计算样本生成的算法流程。首先从 Ω 中任意生成一个初始状态,并通过 MATLAB 中的 GA 工具箱对该初始状态进行求解。如果存在鞍点解,则对其进行归一化处理,并将鞍点处的相对状态轨线根据时间间隔分为 5 个

采样点,分别将这5个采样点处的相对状态和归一化后的解作为新生成的样本存储到样本容器,如图3所示。这样的处理可以充分利用鞍点轨线上的信息,减少样本生成的计算负担。需要注意的是,此处5个采样点的设置并不是固定的,根据问题不同可对其进行调整。

算法1 训练样本生成算法

Alg.1 Generating samples for training

输入:已有样本数量 n ,所需样本数量 N, Ω

输出:包含 N 个样本的样本容器

```

n = 0
while n < N do
    从  $\Omega$  中随机生成初始状态  $x_{pe}$  和  $T_{pe}$ 
    使用 MATLAB 的 GA 工具箱计算  $v$  和  $t_f$ 
    if  $v$  和  $t_f$  存在
        归一化处理  $v$  得到  $\xi$ 
        沿  $\xi$  和  $t_f$  获得最优轨线  $x_{pe}^*(t)$ 
        令  $t = 0$ 
        while  $t < t_f$  do
            保存  $x_{pe}^*(t), T_{pe}, \xi$  和  $t_f - t$  为一个样本
            n = n + 1
            t = t +  $t_f/5$ 
        end while
    end if
end while

```

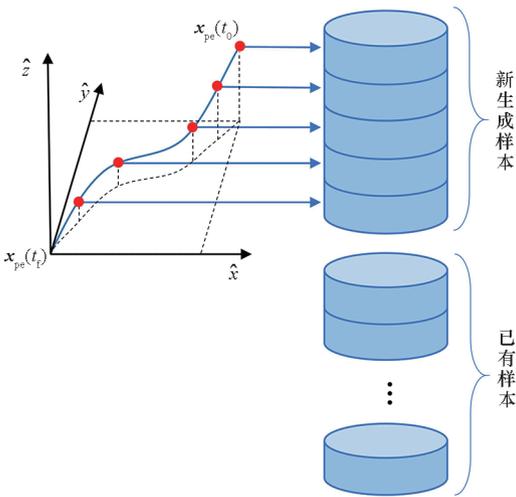


图3 一个初始状态中的5个采样点
Fig.3 Five sampling point in an initial state

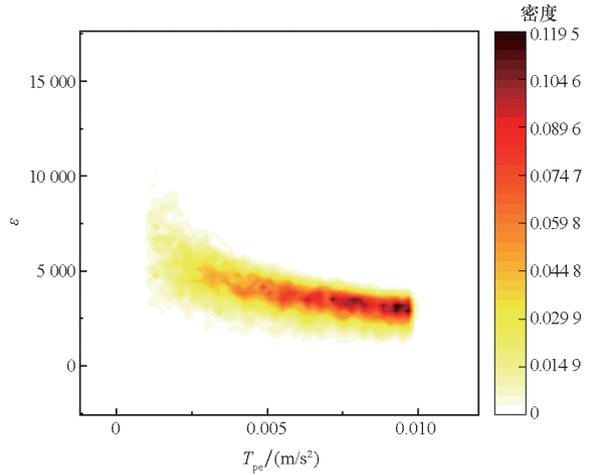
2.3 样本预处理与神经网络训练

2.3.1 样本数据预处理

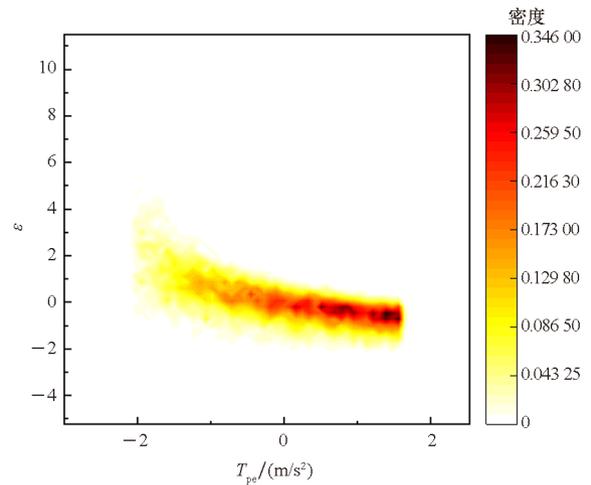
在生成大量的样本数据后,每一个样本都由问题的输入和输出组成,其中输入为 η 、输出为 σ 。

为了避免样本数据量级不同对网络训练产生的影响,提高网络训练的精度,进一步采用基于 Python 的 Scikit-Learn 库对样本集进行数据标准化处理,将每一维度的样本数据减去均值并除以标准差,从而使预处理后的数据均值为0,方差为1。

为展示样本数据的变化情况,图4给出了在预处理前后 T_{pe} 和 ε 核密度分布,从中可以看出,样本数据的相对分布并未发生明显改变,但其整体分布更为集中,且二者之间的数量级更为接近。



(a) 预处理前
(a) Before preprocessing



(b) 预处理后
(b) After preprocessing

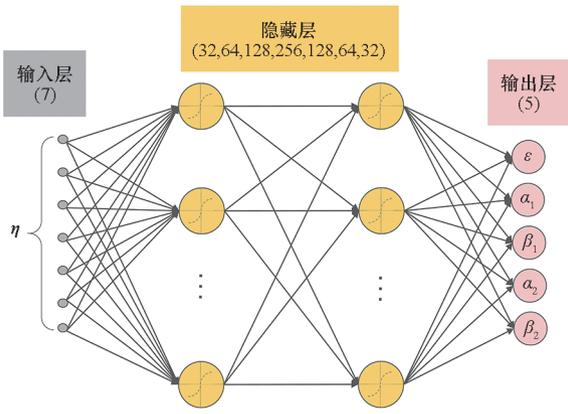
图4 T_{pe} 与 ε 核密度分布

Fig.4 Kernel density distribution of T_{pe} and ε

2.3.2 神经网络训练

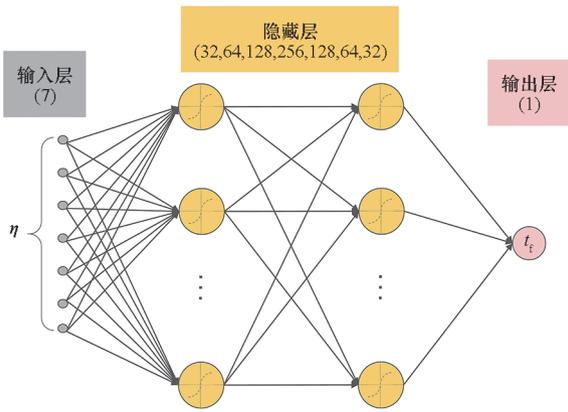
根据物理含义的不同,针对终端协态变量和 t_f 分别设计两个7层全连接网络结构,如图5所示。

选择 ReLU 函数作为隐藏层激活函数,由于输出层节点具备上下限从而选择 Tanh 函数作为激活函数更为合适。将两个神经网络的输出与样



(a) 终端协态变量神经网络

(a) Neural network of terminal costate variable



(b) 终端时刻神经网络

(b) Neural network of t_f

图 5 神经网络结构

Fig. 5 Structure of neural networks

本中的 σ 比较并计算均方误差 (mean-square error, MSE) 作为损失函数进行梯度回传,从而不断调整节点参数,提高网络的拟合效果。经过 100 000 次训练,网络训练结果如图 6 所示,可以看出损失函数变化逐渐接近 0 且逐渐趋于平稳。

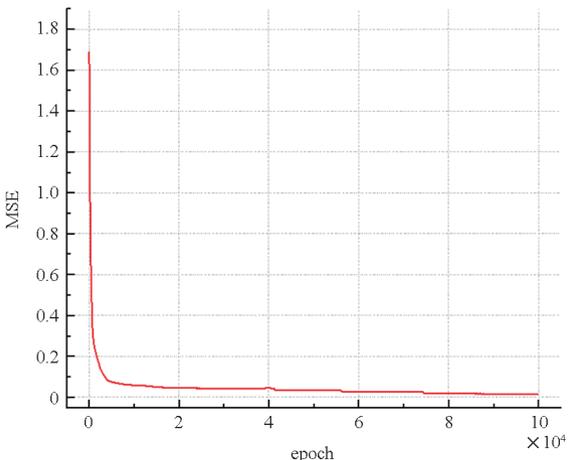


图 6 网络训练过程中的损失函数

Fig. 6 Loss function in network training

3 滚动时域优化求解框架

由于微分对策模型的特点,无论是数值优化算法还是神经网络输出,其结果只能作为一个开环控制解,难以根据博弈态势的变化进行策略调整。为解决这一问题,设计了一种滚动时域优化求解框架,通过将神经网络猜测轨迹与伪谱法结合实现策略的快速更新。

3.1 基于神经网络和伪谱法的策略生成

通过生成样本并对神经网络进行训练,在已知问题初始条件时,通过将初始状态输入到神经网络结构,理论上可以得到一个较为精确的解。但考虑在以交会为博弈目标的航天器追逃问题模型中,神经网络输入和输出参数均较多,训练得到的神经网络可能存在一定的误差,又因为该问题对协态变量值和终端时刻值均非常敏感,因此难以将网络输出直接应用于控制策略,需要对其进行进一步优化。

在已知网络输出时,可以将神经网络输出的猜测解 $\tilde{\sigma}$ 代入到 1.3 节的微分博弈模型中,从而获得追逃双方在博弈过程中的参考轨迹。根据 1.3 节中的模型特点,该参考轨迹表示追逃双方的相对位置和相对速度的变化。通过将这一参考轨迹作为一个初始解,进一步应用 Radau 伪谱法进行优化求解,从而可以得到针对当前状态的精确解,这一过程将基于 GPOPS 工具箱^[18]实现。这一求解过程相较于 GA 或差分进化算法 (differential evolution, DE) 可以大幅减少计算时间,从而为后续根据博弈态势快速更新策略提供条件。

此外,GPOPS 仅输出了节点处的控制变量,而本文所关注的问题均为连续小推力条件下的连续博弈过程,为了获得连续的控制序列还需根据 Radau 伪谱法的 Lagrange 插值原理获得连续的控制变量。

3.2 滚动时域框架内的策略更新

根据上一小节中的算法流程,可以在已知当前状态时,快速得到一个不含反馈结构的开环解。为了应对博弈过程中存在的不确定性,借鉴滚动时域优化的思想,基于当前测量的相对状态 $\tilde{\eta}$,将获得的开环策略应用于接下来的控制时域(即一个周期)内,并不断重复此过程直至满足博弈的终止条件,整体流程如图 7 所示。

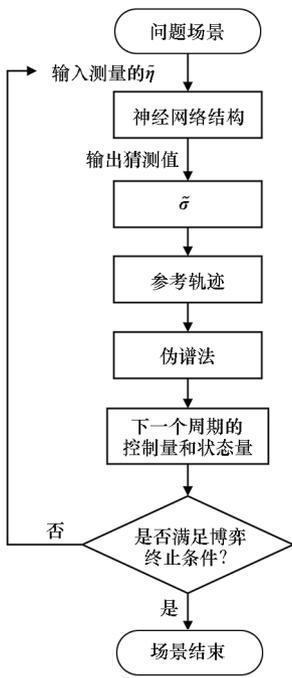


图7 滚动时域优化流程

Fig. 7 Rolling time domain optimization process

4 仿真验证与分析

4.1 算例设置

考虑追踪方在交会博弈过程中测量得到的相对状态信息始终受到噪声干扰,在博弈开始时,追踪航天器和逃逸航天器均位于 GEO 附近(当追逃场景位于其他轨道附近时,应考虑重新生成样本对神经网络进行训练)。在 LVLH 坐标系下,追踪方相对于逃逸方的位置速度以及追逃双方的推力大小设置如表 2 所示。

表 2 仿真算例初始状态

Tab. 2 Initial state of simulation

状态值	追踪方(p)	逃逸方(e)
x/km	-60	-30
y/km	-60	-30
z/km	-60	-30
$v_x/(\text{m/s})$	0.5	0
$v_y/(\text{m/s})$	0.5	0
$v_z/(\text{m/s})$	0.5	0
$T/(\text{m/s}^2)$	0.006	0.003

假设追踪方在滚动时域优化框架内每隔 60 s 进行一次相对状态测量,测量所包含的误差符合正态分布,其均值为真实值,方差取真实值的

1%。每次采样完成后,追踪方根据测量所得的 $\tilde{\eta}$ 通过第 3 节所述算法流程,输出一个周期内的控制序列,并不断进行更新采样直至博弈结束。需要注意的是,这样的算法框架是基于微分对策理论得出的,即假定逃逸方采取了鞍点策略,而当逃逸方采取了其他策略时,该算法所得的控制序列则会偏为保守。

4.2 仿真结果

图 8 中给出了在空间中追踪方相对逃逸方的轨迹变化。图 9 和图 10 则分别为沿坐标轴三个方向相对位置和相对速度的变化。在经过 8 139. 766 s 后,追踪方和逃逸方的位置和速度均符合交会条件,此次博弈过程终止。

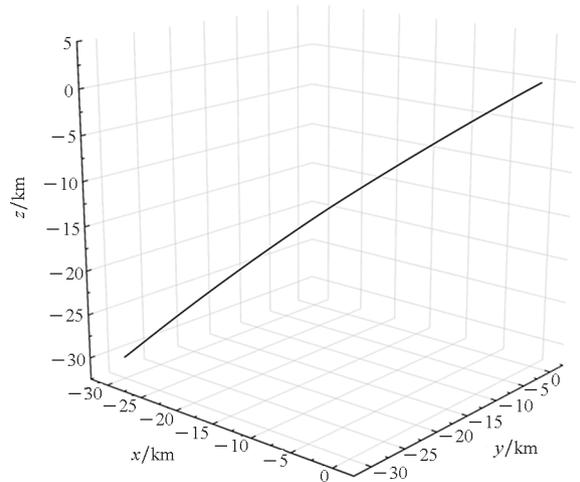


图 8 博弈过程中的相对轨迹

Fig. 8 Relative trajectory during the game

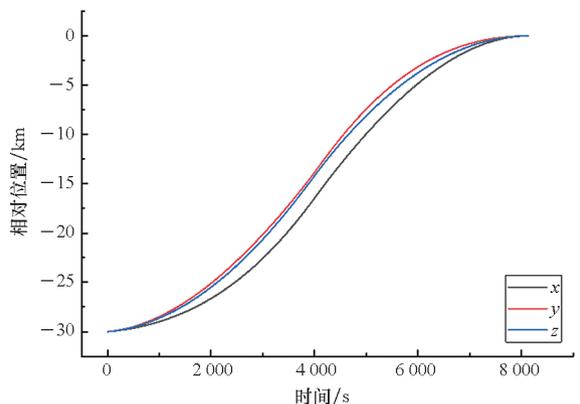


图 9 相对位置变化

Fig. 9 Relative position change

图 11 则对滚动优化过程中每次采样的测量值和真实值进行了对比,可以发现,虽然测量误差存在于整个博弈过程中,但在论文所提算法框架下仍可以完成交会博弈任务。这是因为一方面随着不断接近逃逸方,测量过程中的绝对误差不断

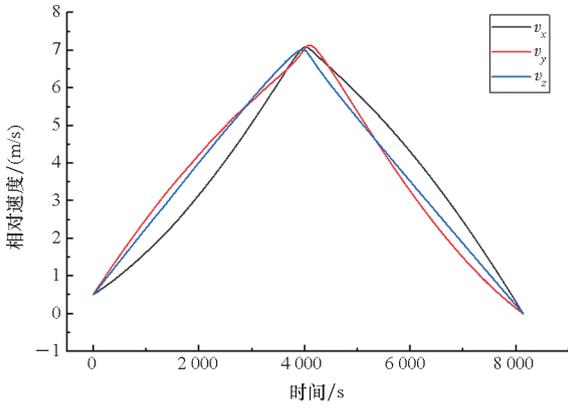


图 10 相对速度变化

Fig. 10 Relative velocity change

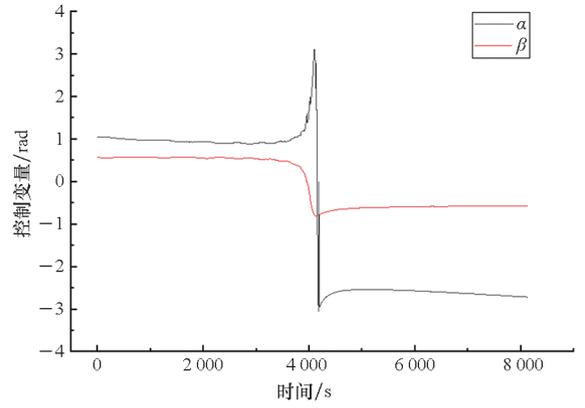
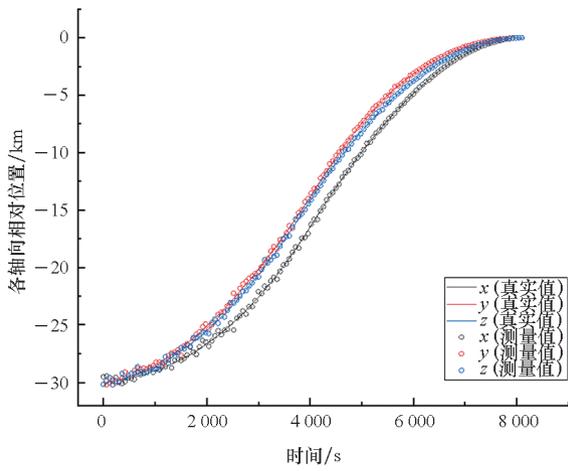


图 12 控制变量

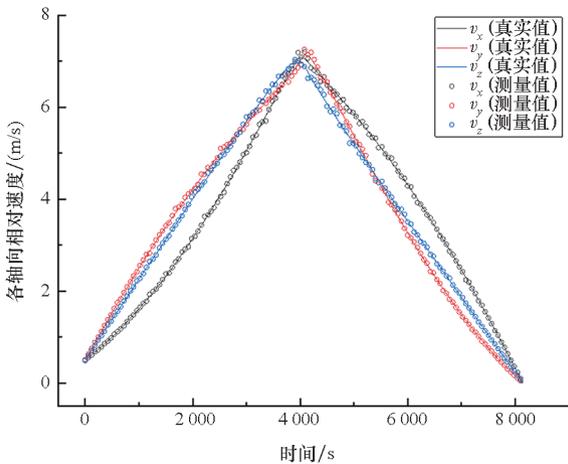
Fig. 12 Control value

减小,另一方面则是通过不断更新测量值从而不断对控制序列进行调整。



(a) 相对位置对比

(a) Comparison of relative positions



(b) 相对速度对比

(b) Comparison of relative velocities

图 11 真实值和采样值对比

Fig. 11 Comparison of real value and sampling value

在经过对 GPOPS 输出的控制变量进行插值后可以得到追踪方的控制序列如图 12 所示。

为了说明在伪谱法中嵌入神经网络的作用,下面将对首次采样的计算结果进行说明。表 3 所示为首次采样时测量所得的相对状态,此时若采用开环策略进行追逃,不仅计算时间较长且会造成较大的终端误差,从而导致任务失败无法完成对目标的交会。表 4 中为在完成首次采样后,神经网络输出值的大小 $\tilde{\sigma}$ 和应用 GA 计算得到开环策略对应的解的大小 σ ,根据表 4 中的结果对比,可以看出测量值对应的神经网络输出和当前真实相对状态对应的开环策略解是较为接近的,从而可以提供较为准确的参考轨迹。

表 3 首次采样结果

Tab. 3 The first sampling result

状态值	测量值	真实值
x/km	-29.513	-30
y/km	-30.184	-30
z/km	-30.158	-30
$v_x/(\text{m/s})$	0.495	0.5
$v_y/(\text{m/s})$	0.504	0.5
$v_z/(\text{m/s})$	0.489	0.5

表 4 神经网络与 GA 计算结果对比

Tab. 4 Comparison of neural networks and GA

问题解	$\tilde{\sigma}$	σ
v_1	-1.637×10^{-4}	-1.729×10^{-4}
v_2	-1.339×10^{-4}	-1.391×10^{-4}
v_3	-1.346×10^{-4}	-1.268×10^{-4}
v_4	0.733 6	0.767 4
v_5	0.351 5	0.339 8
v_6	0.581 6	0.543 7
t_f/s	7 614. 878	8 127. 704

为体现所设计算法的时效性,将本文算法与

文献[11]算法进行了对比。其中,本文计算环境为一台 2.70 GHz、16 GB 的计算机,软件平台为 MATLAB 的 GPOPS 工具箱和 Python3.9。文献[11]计算环境为一台 2.4 GHz、16 GB 的计算机,软件平台为 MATLAB 的 GA 工具箱。计算环境和软件平台差异不大,结果具有可比性。图 13 所示为应用本文算法进行求解的计算耗时,可以看出每次计算的时间消耗为秒级,相较于文献[11]中的分钟级耗时,计算效率具有明显提升。

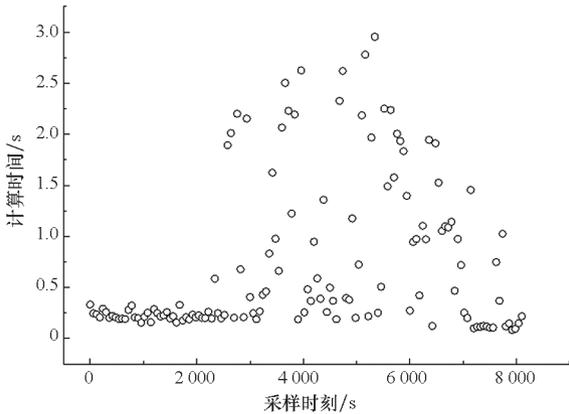


图 13 计算时间

Fig. 13 computing time

5 结论

本文针对航天器自由时域交会型轨道追逃问题,研究了基于滚动时域优化的追逃策略,主要得到以下结论:

1) 所提出的滚动时域优化框架可有效应对追逃过程中的测量误差等不确定性影响;

2) 通过引入神经网络并与伪谱法相结合,可将策略求解计算耗时提升至秒级。

在未来的工作中,可以尝试进一步将该方法应用于逃逸方执行未知机动或逃逸方存在护卫航天器等场景的研究工作中。

参考文献 (References)

[1] 胡海鹰,朱永生,江新华. 美国高轨空间安全发展态势及其关键技术[J]. 空间控制技术与应用, 2022, 48(3): 1-10.
HU H Y, ZHU Y S, JIANG X H. The development trend of high earth orbit space security and key technologies [J]. Aerospace Control and Application, 2022, 48(3): 1-10. (in Chinese)

[2] WEEDEN B, SAMSON V. Global counterspace capabilities[R/OL]. [2023-03-01]. https://swfound.org/media/207350/swf_global_counterspace_capabilities_2022_rev2.pdf.

[3] 周俊峰. 基于微分对策理论的航天器追逃控制方法研究[D]. 哈尔滨: 哈尔滨工程大学, 2021.

ZHOU J F. Research on control method for spacecraft pursuit-evasion based on differential game theory [D]. Harbin: Harbin Engineering University, 2021.

[4] 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.

[5] CARR R W, COBB R G, PACHTER M, et al. Solution of a pursuit-evasion game using a near-optimal strategy [J]. Journal of Guidance, Control, and Dynamics, 2017, 41(4): 841-850.

[6] SHEN H X, CASALINO L. Revisit of the three-dimensional orbital pursuit-evasion game [J]. Journal of Guidance, Control, and Dynamics, 2018, 41(8): 1820-1828.

[7] ZHANG J R, ZHANG K P, ZHANG Y, et al. Near-optimal interception strategy for orbital pursuit-evasion using deep reinforcement learning [J]. Acta Astronautica, 2022, 198: 9-25.

[8] SHI M M, YE D, SUN Z W, et al. Spacecraft orbital pursuit-evasion games with J2 perturbations and direction-constrained thrust[J]. Acta Astronautica, 2023, 202: 139-150.

[9] 常燕,陈韵,鲜勇,等. 机动目标的空间交会微分对策制导方法[J]. 宇航学报, 2016, 37(7): 795-801.
CHANG Y, CHEN Y, XIAN Y, et al. Differential game guidance for space rendezvous of maneuvering target [J]. Journal of Astronautics, 2016, 37(7): 795-801. (in Chinese)

[10] TARTAGLIA V, INNOCENTI M. Game theoretic strategies for spacecraft rendezvous and motion synchronization [C]// Proceedings of the AIAA Guidance, Navigation, and Control Conference, 2016: 0873.

[11] PRINCE E R, HESS J A, COBB R G, et al. Elliptical orbit proximity operations differential games [J]. Journal of Guidance, Control, and Dynamics, 2019, 42(7): 1458-1472.

[12] VENIGALLA C, SCHEERES D J. Delta-V-based analysis of spacecraft pursuit-evasion games [J]. Journal of Guidance, Control, and Dynamics, 2021, 44(11): 1961-1971.

[13] HAFER W T, REED H L, TURNER J D, et al. Sensitivity methods applied to orbital pursuit evasion [J]. Journal of Guidance, Control, and Dynamics, 2015, 38(6): 1118-1126.

[14] 查文中. 单个优势逃跑者的多人定性微分对策研究[D]. 北京: 北京理工大学, 2016.
ZHA W Z. Multi-player qualitative differential games with single superior evader [D]. Beijing: Beijing Institute of Technology, 2016. (in Chinese)

[15] 程林,蒋方华,李俊峰. 深度学习在飞行器动力学与控制中的应用研究综述[J]. 力学与实践, 2020, 42(3): 267-276.
CHENG L, JIANG F H, LI J F. A review on the applications of deep learning in aircraft dynamics and control [J]. Mechanics in Engineering, 2020, 42(3): 267-276. (in Chinese)

[16] CHENG L, WANG Z B, JIANG F H, et al. Fast generation of optimal asteroid landing trajectories using deep neural networks[J]. IEEE Transactions on Aerospace and Electronic Systems, 2020, 56(4): 2642-2655.

[17] CHENG L, WANG Z B, SONG Y, et al. Real-time optimal control for irregular asteroid landings using deep neural networks[J]. Acta Astronautica, 2020, 170: 66-79.

[18] PATTERSON M A, RAO A V. GPOPS-II: a MATLAB software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming [J]. ACM Transactions on Mathematical Software, 2014, 41(1): 1-37.