

一类最佳飞行轨迹的计算^{*}

西北工业大学 王培德 崔桃瑞 侯明

摘 要

本文给出了一种简化的直接多重打靶算法,并讨论了该算法在求解一类最佳飞行轨迹上的应用。数值计算结果表明,本文所给的算法收敛性好,对初值无很高要求,因而避免了在计算之前对问题进行冗繁的解析分析。

关键词 最佳轨迹,非线性规划,直接多重打靶法。

一、引 言

对某项指定的飞行任务,飞行器可采取不同的飞行方案。根据设计计算出的最佳轨迹,可为指定的飞行任务提供最佳飞行方案,以减少运行时间或运行费用等。这对提高飞行效益是很有意义的。

用某些简化模型的方法求解这类问题的缺点是,需要对飞行器的轨迹运动特性有一个很好的理解,而且解的精度往往很低,只能得到次佳解。若利用最佳控制理论中的庞特里雅金(Pontryagin)极值原理,引入哈密顿(Hamilton)函数,便可将最佳轨迹问题化为一个规范的两点边值问题(Two Point Boundary Value Problem,简称TPBVP)^[1]。然而,一类求解TPBVP的算法均对初始解有较高要求。由于TPBVP中协态变量无明显的物理意义,选择好的协态变量初值往往很困难。因此,用这些方法求解一般飞行轨迹优化问题很不方便。

本文根据Bock等^[2]提出的直接多重打靶算法的思想,给出一种适用于求解一类飞行轨迹优化问题的简化算法。由于本文描述的算法是通用的,它可应用于一大类最佳飞行轨迹的计算。

二、问题描述

设飞行器的轨迹运动由下述一组非线性常微分方程描述

$$\dot{x}(t) = f(x(t), u(t), t) \quad (1)$$

$x \in R^l$, $u \in R^q$, $t \in [0, T]$, $T \in R$ 。其中控制变量 $u(t)$ 的变化范围受到约束

$$l_1 \leq u(t) \leq h_1 \quad (2a)$$

状态变量(轨迹) $x(t)$ 的变化范围受到约束

$$l_2 \leq x(t) \leq h_2 \quad (2b)$$

1987年3月10日收到

● 国家自然科学基金资助的课题

初始状态是指定的

$$x(0) = x^0 \quad (3a)$$

终端状态是指定的 (或部分指定)

$$x^f(T) = x^f \quad (3b)$$

其中, $x^f(T) = (x_1(T), x_2(T), \dots, x_{n_d}(T))^T$, $x^f \in R^{n_d}$, $1 \leq n_d \leq l$ 。 n_d 是指定终端状态变量的个数。

轨迹优化的目的就是寻求一个容许控制律 $u(t)$, 使期望的性能指标

$$J = \int_0^T L(x(t), u(t), t) dt \quad (4)$$

达到最小值 (不失一般性)。所谓容许控制是指所有满足约束式 (1)~式 (3b) 的控制律。性能指标式 (4) 中终端时刻 T 可以是固定的, 也可以是自由的。由于大多数飞行器最佳轨迹问题中 T 是自由的, 故本文只讨论后一种情况。至于 T 为固定这种相对简单的情形, 根据本文容易得出对应的处理方法。

对上述非线性最佳轨迹问题, 若不求助于数值计算, 即使要找出一个容许控制也是相当困难的, 故有必要寻求一个有效的数值解法。

三、简化的Mayer问题

直接研究自由终端时间问题是不方便的。另外, 在 Bock 等的算法中, 对由式 (4) 表示的积分性能指标的直接处理方式比较复杂, 对应梯度的计算量也很大。本文将上节描述的问题先转换成固定时间端点的 Mayer 问题, 再利用直接多重打靶法处理。这样将使得本文算法简明, 目标函数的梯度计算量减小。

为此, 引入新的时间变量 $\tau \in [0, 1]$ 。并设 T 为可变参数, 且在一定范围内变化 $l_3 \leq T \leq h_3$ 。定义 $t = T\tau$, 及 $\frac{dt}{d\tau} = T$ 和 $\frac{dT}{d\tau} = 0$ 。同时引入增广状态变量 x_{l+1} 及状态方程

$$\dot{x}_{l+1}(\tau) = T \cdot L(x(\tau), u(\tau), T\tau), \quad x_{l+1}(0) = 0$$

(对最短时间问题, 函数 $L = 1$, 不必引入 x_{l+1} 及对应的状态方程)。

容易证明, 上节描述的最佳轨迹问题与下述固定终端时间的 Mayer 问题等价

$$\min f_1(x_n(1), T) \quad (5a)$$

状态方程

$$\dot{x}(\tau) = Tf(x(\tau), u(\tau), T\tau) = F(x(\tau), u(\tau), T, \tau) \quad (5b)$$

边界条件

$$\begin{cases} x(0) = x^0 \\ x^f(1) = x^f \end{cases} \quad (5c)$$

上、下界约束

$$\begin{cases} l_1 \leq u(\tau) \leq h_1 \\ l_2 \leq x(\tau) \leq h_2 \\ l_3 \leq T \leq h_3 \end{cases} \quad (5d)$$

其中, $x^f(1) = (x_1(1), x_2(1), \dots, x_{n_d}(1))^T$; $x \in R^n$, $u \in R^q$, $1 \leq n_d \leq n$ 。对最

短时间问题, 式 (5a) 中目标函数 $f_1 = T$, 状态方程 (5b) 的维数 $n = l$; 对一般问题, $f_1 = x_n(1)$, $n = l + 1$ 。状态方程以 τ 为时间变量 $\tau \in [0, 1]$ 。

用直接多重打靶法处理此 Mayer 问题是方便的, 目标函数的处理也将得到简化。

四、用直接多重打靶法建立非线性规划问题

由于有状态方程 (5b) 的约束, 要直接求解上述 Mayer 问题仍比较困难。

所谓多重打靶法^[8], 是将时间区间划分为若干段, 在段与段相交的节点上给出一些猜测值, 作为状态变量和协态变量在这些节点上的初始估计值。假定已知各段的控制律, 将描述 TPBVP 的常微分方程从各段的初始节点积分到终端节点, 判断这些积分值是否与对应终端节点处估计值相等 (或称 '匹配 (Match)')。若不相等, 则依一定规则修正这些估计值, 使其逐渐在终端节点处匹配。

在直接多重打靶算法中, 首先需将待求的控制律 $u(\cdot)$ 参数化。参数化的控制不但要使状态变量 (由常微分方程初值问题描述) 在终端节点处匹配, 且要使目标函数达到最小值。这些条件若不满足, 将不断修正各节点上状态变量的猜测值以及参数化了的控制律。这样就可把上节所述 Mayer 问题转化为一般的非线性规划 (Nonlinear Programming, 简记 NLP) 问题。以下是转化步骤:

(1) 将固定时间区间 $[0, 1]$ m 等分, 得到节点: $\tau_i = \frac{i}{m}$, $i = 0, 1, \dots, m$

(2) 控制律参数化: 引入一组向量 $P_i \in R^q$, $i = 0, 1, \dots, m$ 。定义

$$u(\tau) = u_i(P_0, P_1, \dots, P_m), \quad \tau \in [\tau_i, \tau_{i+1}]。$$

其中, $u_i(\cdot)$ 是插值函数。本文取线性插值函数, 即

$$u_i(\tau) = P_i + C_i(\tau - \tau_i) \quad \tau \in [\tau_i, \tau_{i+1}], \quad i = 0, 1, \dots, m-1$$

其中 $C_i = (P_{i+1} - P_i) / (\tau_{i+1} - \tau_i)$ $i = 0, 1, \dots, m-1$

(3) 初值问题建立: 选一组向量 $S_i \in R^n$, 作为节点 τ_i 上状态变量 $x(\tau_i)$ 的估计值, $i = 0, 1, \dots, m$ 。可形成 m 个初值问题

$$\dot{x}(\tau) = F(x(\tau), u(\tau), T, \tau) \quad \tau \in [\tau_i, \tau_{i+1}]$$

初值 $x(S_i, u_i, T, \tau_i) = S_i \quad i = 0, 1, \dots, m-1$ 。

(4) NLP 问题构成

$$\min f_1(S_m^*, T) \quad (6a)$$

满足

$$\text{匹配条件} \quad x(S_i, u_i, T, \tau_{i+1}) - S_{i+1} = 0, \quad i = 0, 1, \dots, m-1 \quad (6b)$$

边界条件

$$\begin{cases} S_0 = x^0 \\ S_m^f = x^f \end{cases} \quad (6c)$$

上、下界约束

$$\left. \begin{aligned} l_1 \leq P_i \leq h_1 & \quad i = 0, 1, \dots, m \\ L_2 \leq S_i \leq H_2 & \quad i = 0, 1, \dots, m \\ l_3 \leq T \leq h_3 & \end{aligned} \right\} \quad (6d)$$

至此, 得到了一个与原问题等价的 NLP 问题。优化变量为 $Y = (S_0^f, S_1^f, \dots, S_m^f, P_0^f, P_1^f, \dots, P_m^f, T)^T$, 目标函数为 f_1 , 约束为式 (6b)~式 (6d)。与问题式 (5) 类

似, 这里, 对最短时间问题 $f_1 = T$; 对一般问题 $f_1 = S_m^*$, S_m^* 是 $x_m(1)$ 的估计值。 $S_m^* = x^f(1)$ 。

五、NLP 问题的求解

原则上, 可采用任何一个 NLP 算法迭代求解问题式 (6), 但序列二次规划算法^[4] 看来最合适。一方面, 许多比较研究表明^[6], 该算法是目前最有效的 NLP 算法之一, 而且该算法对数值微分不敏感^[6] (用数值微分逼近式 (6b) 的梯度一般是无法避免的)。另一方面, 由于序列二次规划算法中下降方向是一个二次规划子问题的解, 而问题式 (6) 的二次规划子问题的维数可以大大减小, 这就使得该子问题的求解变得容易。本文不打算对序列二次规划算法进行详细地讨论。

六、数值计算实例

我们用本文给出的算法计算了两个最佳飞行轨迹。第一个例引自文献[7], 研究某气垫船模型的最短时间飞行问题。第二个例是一实际问题, 引自文献[8], 研究某阿波罗型飞船的推力再入最佳轨迹。文献[7]和文献[8]分别用配置赫密特法 (Collocated Hermit Method) 和多重打靶法给出了这两个例的计算结果, 而这两种算法都是为求解 TPBVP 设计的。

例 1 气垫船最短时间飞行轨迹

(1) 问题描述 某低速运动着的气垫船受一推力 F 控制, 其幅值 A 只能取 F_m 或零, 推力角 α 是连续可控的。气垫船的质量 m 在一定范围内可调整。设摩擦阻力 $f = 0$ (理想情况), 该气垫船的运动轨迹可由以下一组常微分方程描述 (参考坐标系见图 1)

$$\left. \begin{aligned} \dot{x}(t) &= v_x(t) \\ \dot{y}(t) &= v_y(t) \\ \dot{v}_x(t) &= \frac{A}{m} \cos \alpha(t) \\ \dot{v}_y(t) &= \frac{A}{m} \sin \alpha(t) \end{aligned} \right\} \quad (7)$$

求出控制律 $\alpha(t)$, 它以最短时间 T , 将气垫船由初始状态 $x(0) = y(0) = v_y(0) = 0$, $v_x(0) = 5$ 导引到终端状态 $x(T) = y(T) = v_y(T) = 5$, $v_x(T) = 0$ 。

(2) 问题求解 假定取参数 $\frac{A}{m} = 3$, 并取状态、控制、时间参数变量的上、下界约束分别为: $(0, 0, 0, -10)^T \leq (x, y, v_x, v_y)^T \leq (30, 30, 20, 10)^T$, $-\pi \leq \alpha \leq \pi$, $0 \leq T \leq 15$ 。初始估计 (节点数取为 10) 为: 控制 $\alpha^0(t) = 0$, 时间参数 $T^0 = 2$, 节点上的状态为初始状态和终端状态之间的线性插值。用本文算法计算, 经 22 步迭代, 可得到解如图 2、图 3 所示, 最短时间 $T_{\min} = 2.464$ 。若其它条件不变, 取 $T^0 = 5$,

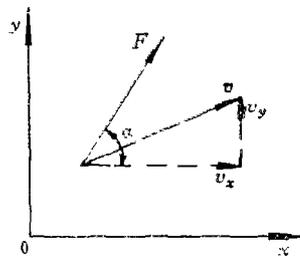


图 1 气垫船轨迹坐标系

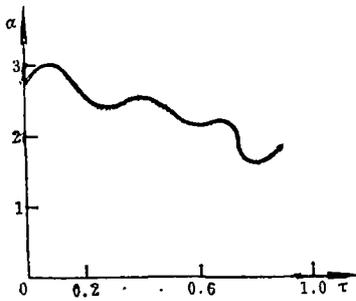


图 2 气垫船最佳控制

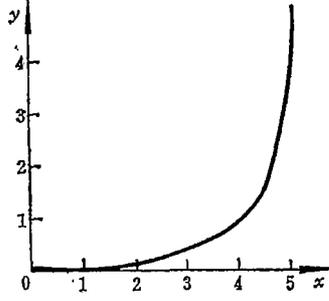


图 3 气垫船最佳轨迹

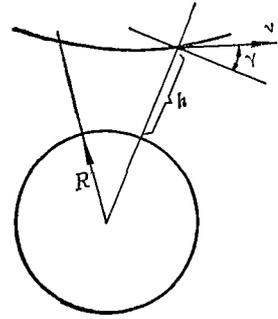


图 4 再入轨迹坐标系

经 22 步迭代，解的精度也很好，此时 $T_{\min} = 2.529$ 。

例 2. 飞船举力再入最佳轨迹

(1) 问题描述 某阿波罗飞船通过大气层的运动轨迹满足下列一组常微分方程(参考坐标系见图 4)

$$\left. \begin{aligned} \dot{v} &= -\frac{S\rho(\xi)v^2}{2m} C_D(u) - \frac{g \sin \gamma}{(1+\xi)^2} \\ \dot{\gamma} &= \frac{S\rho(\xi)v}{2m} C_L(u) + \frac{v \cos \gamma}{R(1+\xi)} - \frac{g \cos \gamma}{v(1+\xi)^2} \\ \dot{\xi} &= \frac{v \sin \gamma}{R} \end{aligned} \right\} \quad (8)$$

其中，状态变量 v 、 γ 、 ξ 分别为速度、航迹角和标称高度 $\left(\frac{h}{R}\right)$ ，减速锥角 u 是控制变量，允许变化范围是 $|u| \leq \frac{\pi}{2}$ 。大气密度函数 $\rho(\xi) = \rho_0 e^{-\beta R \xi}$ ；阻力系数 $C_D(u) = C_1 - C_2 \cos u$ ；升力系数 $C_L(u) = C_3 \sin u$ 。有关参数值略。求出最佳控制 $u(t)$ ，它可使飞船从初始状态 $v(0) = 0.11 (\times 10^8 \text{m/s})$ ， $\gamma(0) = -8.1^\circ \frac{\pi}{180^\circ}$ ， $\xi(0) = \frac{4}{R} \times (h(0) = 1.22 \times 10^5 \text{m})$ ，达到终端状态 $v(T) = 0.27$ ， $\gamma(T) = 0$ ， $\xi(T) = \frac{2.5}{R}$ 。同时要使对流加热量最小

$$\min J = \int_0^T 10v^3 \sqrt{\rho(\xi)} dt$$

终端时间 T 是自由的。

(2) 问题求解 用本文算法求解此问题，引入增广状态变量 $\eta(t)$ 及微分方程

$$\dot{\eta}(t) = 10v^3 \sqrt{\rho(\xi)} \quad \eta(0) = 0$$

取上、下界约束为 $-\pi/2 \leq u \leq \pi/2$ ， $200 \leq T \leq 500$ ， $(0.1, -0.2, 0, 0)^T \leq (v, \gamma, \xi, \eta)^T \leq (0.5, 0.1, 0.03, 0.05)^T$ ；初始估计(节点数取为 10)为 $u^0(t) = 0$ ， $T^0 = 230$ ， $\eta^0(T) = 0.01$ ，节点上的状态变量值取为初始状态和终端状态之间的线性插值。经 20 步迭代，可得到最佳轨迹如图 5~图 7 所示，图中 ϵ_m 表示匹配条件满足的精度范围。为了能看出收敛过程，图 8 给出了最佳控制以及中间迭代结果。性能指标 $J =$

2.4967×10^{-2} , 终端时间 $T = 226.3s$ 。考虑到刚开始的控制对性能指标及匹配条件影响甚微, 在 $\tau \in [0, 0.1]$ 段上可采用常值修正控制 $u(\tau) = 1.5$ 。此时性能指标变为 $J = 2.53 \times 10^{-2}$ 。

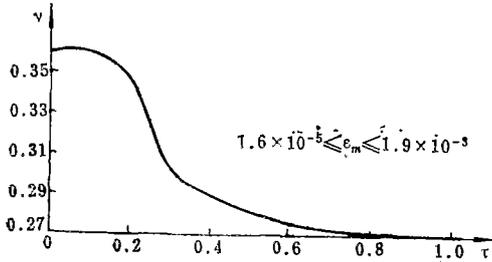


图 5 最佳飞行速度

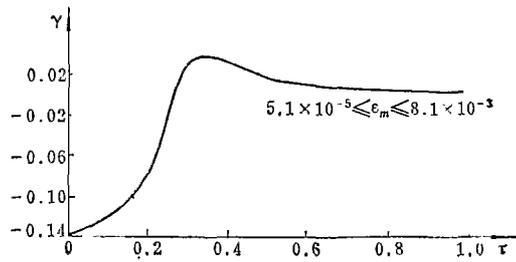


图 6 最佳航迹角

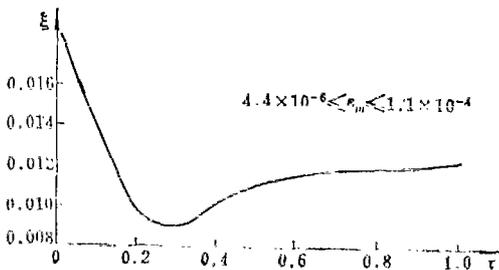


图 7 最佳飞行高度 (标称)

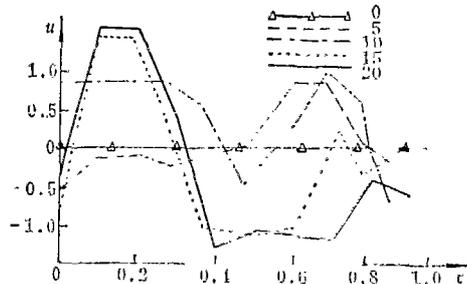


图 8 最佳控制及中间迭代结果

数值计算实验表明, 最佳轨迹问题中上、下界约束值一般对本文算法的收敛性影响不大。这些值选取的依据是对应变数或参数的可能变化范围, 若对此知识甚少, 可选取较宽的界值。从以上两例可看出, 状态变量和控制变量的初值选取是容易的、直观的。数值实验表明算法仅对例 2 时间参数 T 初值的选取比较敏感。

七、结 束 语

本文研究了一种简化的直接多重打靶算法及其在计算最佳飞行轨迹中的应用。算例计算表明, 该算法收敛性好, 解的精度也较高, 对初始估计无苛刻要求。

如果与文献[7、8]的算法进行比较, 可大致得出如下结论: 相对配置赫密特法, 本文算法的精度与其相当或稍高, 但计算量较大; 相对多重打靶法, 本文算法的精度低些, 但已满足一般工程要求。本文算法是一种直接法, 其最大优点在于算法的通用性, 使用方便性, 避免将问题化为 TPBVP 时必须进行的解析分析, 算法初值的选取相对也容易得多。这是通常基于求解 TPBVP 的算法所不能比拟的。

还应该指出, 稍加修改后, 本文给出的算法将可以处理更复杂些的边界约束, 如

$$r_0(x(0)) = 0 \quad \text{和 (或)} \quad r_T(x(T)) = 0$$

从算法原理上看, 进一步将本文算法推广到能处理更一般的约束

$$r(x(t), u(t), t) \geq 0$$

应该没有任何方法上的困难。而这类约束在载人飞行器的最佳轨迹问题中会经常遇到。

参 考 文 献

- [1] Bryson, A. E. and Ho, Y. C., Applied Optimal Control. John Wiley and Son, New York, 1975
- [2] Bock, H. G. and Plitt, K. J., A Multiple Shooting Algorithm for Direct Solution of Optimal Control Problems. Proceeding of the 9th IFAC Conference, 1984
- [3] Deuffhard, P., Recent Advances in Multiple Shooting Techniques. in Computational Techniques for Ordinary Differential Equations, Academic Press Inc, New York, 1980
- [4] 席少霖, 约束变尺度方法, 运筹学杂志, 第四卷, 第一期, 1985
- [5] Schittkowski, K., The Current State of Constrained Optimization Software. in Nonlinear Optimization 1981, eds. by Powell, M. J. D., 1982
- [6] Schittkowski, K., The Nonlinear Programming Methods of Wilson, Han and Powell with an Efficient Implementation with Linear Least Squares Subproblems, Numer. Math., 38, 1981
- [7] Dickmanns, E. D., Flight Trajectory Optimization and Numerical Methods Lecture Series, at Beijing Institute of Aeronautics and Astronautics, 1986
- [8] Store, J. and Bulirsch, R., Introduction to Numerical Analysis, Springer-Verlag New York Inc., 1980

NUMERICAL CALCULATION OF A CLASS OF OPTIMAL FLIGHT TRAJECTORIES

Wang Peide, Cui Taorui and Hou Ming

(Northwestern Polytechnic University)

Abstract

A simplified direct multiple shooting algorithm is presented in this paper. The algorithm is developed to solve a class of optimal trajectory problems with assigned initial state variables, assigned (or partly assigned) terminal state variables, free terminal time and bound constraint both on state and control variables. First, this class of problems is transformed to a Mayer problem with fixed terminal time. Then, a nonlinear programming problem is formed from the Mayer problem using direct multiple shooting technique in the light of Bock and Plitt's work.

Satisfactory numerical results are obtained when an implementation of the presented algorithm is used to solve two optimal flight trajectory problems. The first problem is to minimize the flight time of a hovercraft. The second one is a practical problem in space travel. It is to minimize the total stagnation point convective heating per unit area.

Numerical calculations show that the algorithm presented in this paper has a good convergence and no strict demands for initial guess in dealing with a flight trajectory problem. So compute a flight trajectory problem.

Key words optimal trajectory, nonlinear programming, direct multiple shooting algorithm.