

闭路制导

姓名: 鲁显彻

学号: 2022300707

闭路制导大作业

基本概念:

- 需要速度: 火箭在当前位置关机, 经过自由段和再入段飞行而命中目标的速度
- 虚拟目标: 以需要速度为初值, 椭圆轨道与地球表面的交点. 使用虚拟目标替代实际目标可以略去再入段弹道的计算.

基本思路:

1. 根据虚拟目标和导弹的经纬度信息求解它们的位置矢径
2. 根据位置矢径求解过这两点的椭圆轨道参数, 过两点的椭圆轨道有无数条, 添加限制条件, 比如在目标处的速度倾角, 使得只有一条椭圆轨道满足要求
3. 求解椭圆轨道参数和导弹到目标的飞行时间
4. 考虑地球自转, 将飞行时间带入到目标矢径的旋转矩阵获取新的目标位置, 再次计算椭圆轨道参数, 迭代计算, 直到椭圆参数近似不变
5. 根据椭圆轨道参数计算导弹的需要速度大小和真近点角, 将需要速度转化到地心惯性坐标系中
6. 控制导弹发动机推力方向与需要速度方向相同
7. 当需要速度大小为零时发动机关机

闭路制导原理

求解需要速度

设目标和导弹真近点角的插值为 β

$$\beta = \cos^{-1} \frac{\vec{r}_T \cdot \vec{r}_M}{|\vec{r}_T| |\vec{r}_M|}$$

根据开普勒轨道理论

$$\begin{aligned} r_M &= \frac{p}{1 + e \cos(\theta_T - \beta)} & \theta_T &= \theta_M + \beta \\ r_T &= \frac{p}{1 + e \cos(\theta_T)} \end{aligned}$$

消去 e

$$e = \left(\frac{p}{r_T} - 1 \right) / \cos(\theta_T) \quad (1)$$

$$\begin{aligned}
r_M &= \frac{p}{1 + \left(\frac{p}{r_T} - 1\right) \cos(\theta_T - \beta) / \cos(\theta_T)} \\
&= \frac{p}{1 + \left(\frac{p}{r_T} - 1\right) (\cos \beta + \sin \beta \tan \theta_T)}
\end{aligned}$$

速度倾角 γ 与真近点角 θ 的关系

$$\begin{aligned}
\tan \theta &= \frac{\sin \theta}{\cos \theta} \\
&= \frac{v_r / e \sqrt{\frac{p}{\mu}}}{\left(\frac{p}{r} - 1\right) / e} \\
&= \frac{pv \sin \gamma}{\left(\frac{p}{r} - 1\right) H} \quad H = rv \cos \gamma \\
&= \frac{p \tan \gamma}{p - r} \tag{2}
\end{aligned}$$

添加在目标处的速度倾角约束 γ_T

$$\begin{aligned}
r_M &= \frac{p}{1 + \left(\frac{p}{r_T} - 1\right) \left(\cos \beta + \sin \beta \frac{p \tan \gamma_T}{p - r_T}\right)} \\
p &= \frac{r_M (1 - \cos \beta)}{1 - \frac{r_M}{r_T} (\cos \beta + \sin \beta \tan \gamma_T)}
\end{aligned}$$

将 p 代回式(2)计算 θ_T , 代回式(1)计算 e

计算偏近点角 E

$$\tan \left(\frac{E}{2} \right) = \sqrt{\frac{1-e}{1+e}} \tan \left(\frac{\theta}{2} \right)$$

计算平近点角

$$M = E - e \sin E$$

导弹的飞行时间:

$$t = \sqrt{\frac{a^3}{\mu}} (M_T - M_M)$$

考虑地球自转, 经过导弹飞行时间 t 之后目标新的位置(地心惯性坐标系下)

$$r'_T = R_z(\omega_E t) r_T$$

其中, ω_E 为地球自转角速度

将 r'_T 重新代回第一步, 迭代求解轨道参数, 最终获得轨道参数 a, e, θ_M, t

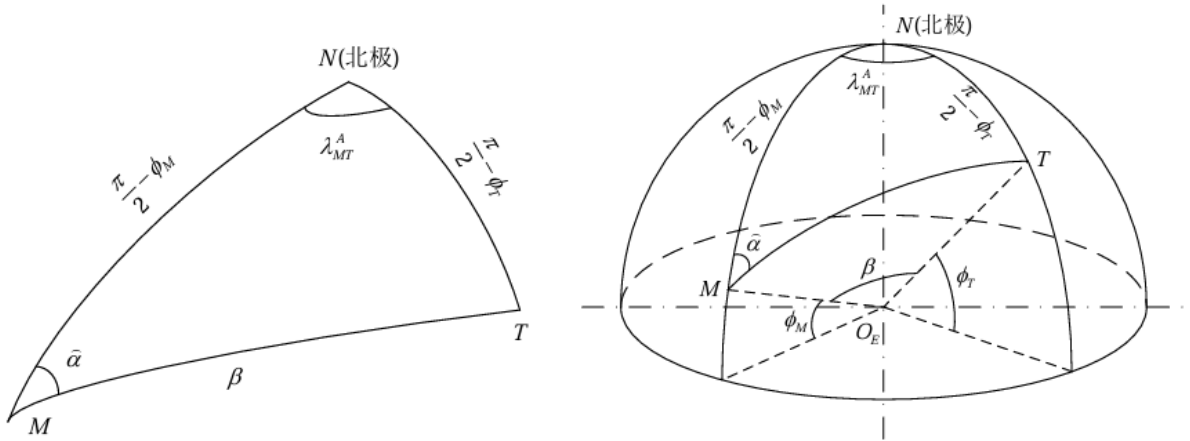
根据轨道参数计算需要速度

$$V_R = \sqrt{\frac{\mu}{p} (1 + e^2 + 2e \cos \theta_M)}$$

需要速度倾角

$$\gamma_R = \tan^{-1}\left(\frac{p - r_M}{p} \tan(\theta_T - \beta)\right)$$

坐标转换



导弹和目标的纬度

$$\phi_M = \sin^{-1}\left(\frac{r_{Mz}}{|\vec{r}_M|}\right)$$

$$\phi_T = \sin^{-1}\left(\frac{r_{Tz}}{|\vec{r}_T|}\right)$$

导弹和目标的经度

$$\lambda_M = \tan^{-1}\left(\frac{r_{My}}{r_{Mx}}\right)$$

$$\lambda_T = \tan^{-1}\left(\frac{r_{Ty}}{r_{Tx}}\right)$$

由球面几何, 方位角:

$$\sin \alpha = \frac{\sin|\lambda_T - \lambda_M| \cos(\phi_T)}{\sin \beta}$$

$$\cos \alpha = \frac{\sin \phi_T - \cos(\beta) \sin(\phi_M)}{\cos(\phi_M) \sin(\beta)}$$

$$\alpha = \begin{cases} \sin^{-1}(\sin \alpha) & \cos \alpha > 0, |\sin \alpha| < |\cos \alpha| \\ \pi \operatorname{sgn}(\sin \alpha) - \sin^{-1}(\sin \alpha) & \cos \alpha < 0, |\sin \alpha| < |\cos \alpha| \\ \operatorname{sgn}(\sin \alpha) \cos^{-1}(\cos(\alpha)) & |\sin \alpha| > |\cos \alpha| \end{cases}$$

当地北东坐标系

- 原点: 任意一点M
- y轴: 地心指向M
- x轴: 过M点子午面内与y轴垂直指向北极

当地北东坐标系下的速度分量

$$\begin{bmatrix} V_R \cos \gamma \cos \alpha \\ V_R \sin \gamma \\ V_R \cos \gamma \sin \alpha \end{bmatrix}$$

当地北东坐标系到地心惯性坐标系的转换矩阵

$$L = R_z(-\lambda_M)R_y(\phi_M)$$

地心惯性坐标系下的需要速度矢量

$$\vec{V}_R = L \cdot \begin{bmatrix} V_R \cos \gamma \cos \alpha \\ V_R \sin \gamma \\ V_R \cos \gamma \sin \alpha \end{bmatrix}$$

推力方向

待增速度

$$\vec{V}_g = \vec{V}_R - \vec{V}$$

推力方向

$$\hat{P} = \frac{\vec{V}_g}{|\vec{V}_g|}$$

MATLAB仿真验证

初始条件

$$\begin{aligned} r_T &= \begin{bmatrix} 1.289983766295003\text{e} + 06 \\ -5.359931448185626\text{e} + 06 \\ 3.194024762781843\text{e} + 06 \end{bmatrix} \text{m} \\ r_{M0} &= \begin{bmatrix} 1164000 \\ -5507000 \\ 3258000 \end{bmatrix} \text{m} \\ v_{M0} &= \begin{bmatrix} 1337 \\ 743 \\ 1029 \end{bmatrix} \text{m/s} \end{aligned}$$

导弹参数

$$\begin{aligned} P &= 20000 \text{ N} \\ m_0 &= 500 \text{ kg} \\ I_{sp} &= 2842 \text{ s} \end{aligned}$$

火箭运动微分方程组

$$\begin{aligned}\frac{d\vec{r}}{dt} &= \vec{v} \\ \frac{d\vec{v}}{dt} &= \frac{1}{m}\vec{P} - \frac{\mu}{|\vec{r}|^3}\vec{r} \\ \frac{dm}{dt} &= -\frac{1}{I_{sp}g_0}|\vec{P}| \qquad g_0 = 9.81 \text{ m/s}^2\end{aligned}$$

程序结构, 完整程序见附件

```

% Rocket.m
% 火箭对象
classdef Rocket < handle
    properties
        P_mag
        I_sp
        t_burn
        omegaE % 地球自转角速度，设置为0忽略自转
        rocoder % 记录数据
        .....
    end
    methods
        function obj = Rocket(t0, dt, tf)
            % 初始化对象
        end
        function value = HitGround(~, ~, y)
            % 判断是否打击地面
        end
        function dydt = RocketDynamics(obj, t, y, r_T0, gama_T)
            % 导弹动力学微分方程，包含制导部分，输入目标初始值
        end
        function V_R = RocketGuidance(obj, r_M, r_T, gama_T)
            % 获得导弹所需速度大小与俯仰角（航迹角）
        end
        function [lambda, phi] = func4(~, r)
            % 输入：位置矢量
            % 输出：经纬度（球面坐标系）
        end
        function alpha = func3(obj, r_T, r_M)
            % 输入：目标和导弹位置矢量，输出：目标和导弹之间的方位角
        end
        function [V_R_mag, gama_R] = func2(obj, r_M, r_T, gama_T)
            % 输入：导弹和目标的失径，在目标点的速度倾角
            % 输出：考虑地球自转，导弹点的速度大小和速度倾角
            % 使用迭代求解
        end
        function [e, p, theta_T, t, beta] = func1(obj, r_M, r_T,
            gama_T, t)
            % 输入：导弹和目标位置失径，在目标点的速度倾角
            % 输出：过这两点的轨道的轨道参数，导弹到目标的运动时间=

```

```
        end
    end
end
```

MATLAB

```
% test_rocket.m
% 测试能否正确解算出需要速度
% 通过与预设轨道参数对比

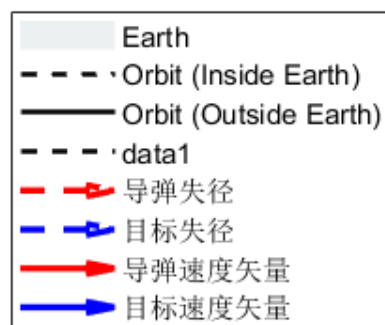
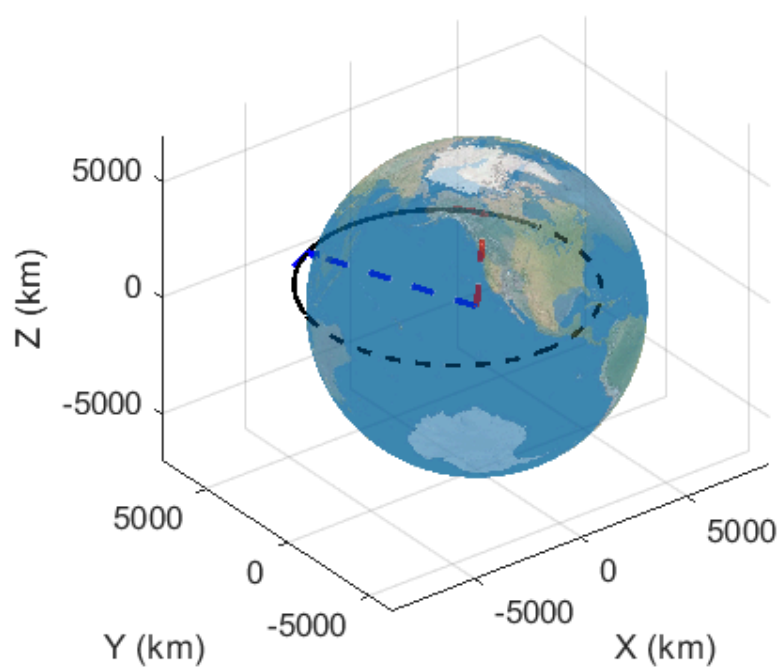
% 详细见附件
```

运行结果

TXT

```
预设：
V_M: 6.84 km/s gama_M: 14.06 deg
lambda_M: 50.00 deg phi_M: 0.00 deg
迭代求解：
t: 1364.55, p: 5548.17
t: 1371.82, p: 5560.75
t: 1372.37, p: 5561.70
t: 1372.41, p: 5561.77
t: 1372.41, p: 5561.78
t: 1372.41, p: 5561.78
t: 1372.41, p: 5561.78
解算需要速度：
V_R: 6.94 km/s gama_R: 13.38 deg
lambda_M: 50.00 deg phi_M: 0.00 deg
方位角：
alpha: 90.00 deg
V_M预设：
    -4.0156
     5.5385
         0
V_R解算：
    -4.1412
     5.5725
     0.0000
```

Satellite Orbit in ECI Frame




```
% main.m
clear; clc; close all;

m_M0 = 500; % kg
r_M0 = [1164; -5507; 3258]; % km
v_M0 = [1.337; 0.743; 1.029]; % km/s
r_T = [1289.9838; -5359.9314; 3194.0248]; % km
gama_T = deg2rad(-55.5302);
y0 = [r_M0; v_M0; m_M0];
t0 = 0; dt = 0.005; tf = 500; % s

rocket = Rocket(t0, dt, tf);
ode = @(t, y) rocket.RocketDynamics(t, y, r_T, gama_T);
event = @(t, y) rocket.HitGround(t, y);
[t, y] = ode_EPC(t0, dt, tf, y0, ode, event);

r_M = y(1:3, 1:length(t)); % 导弹位置
v_M = y(3:6, 1:length(t)); % 导弹速度
P = rocket.recoder.P(1:3, 1:length(t)); % 推力方向
r_rel = rocket.recoder.r_rel(1:length(t)); % 相对位置
r_T = rocket.recoder.r_T(1:3, 1:length(t)); % 目标位置矢量
V_R = rocket.recoder.V_R(1:3, 1:length(t)); % 目标速度矢量

% 绘制图像
```

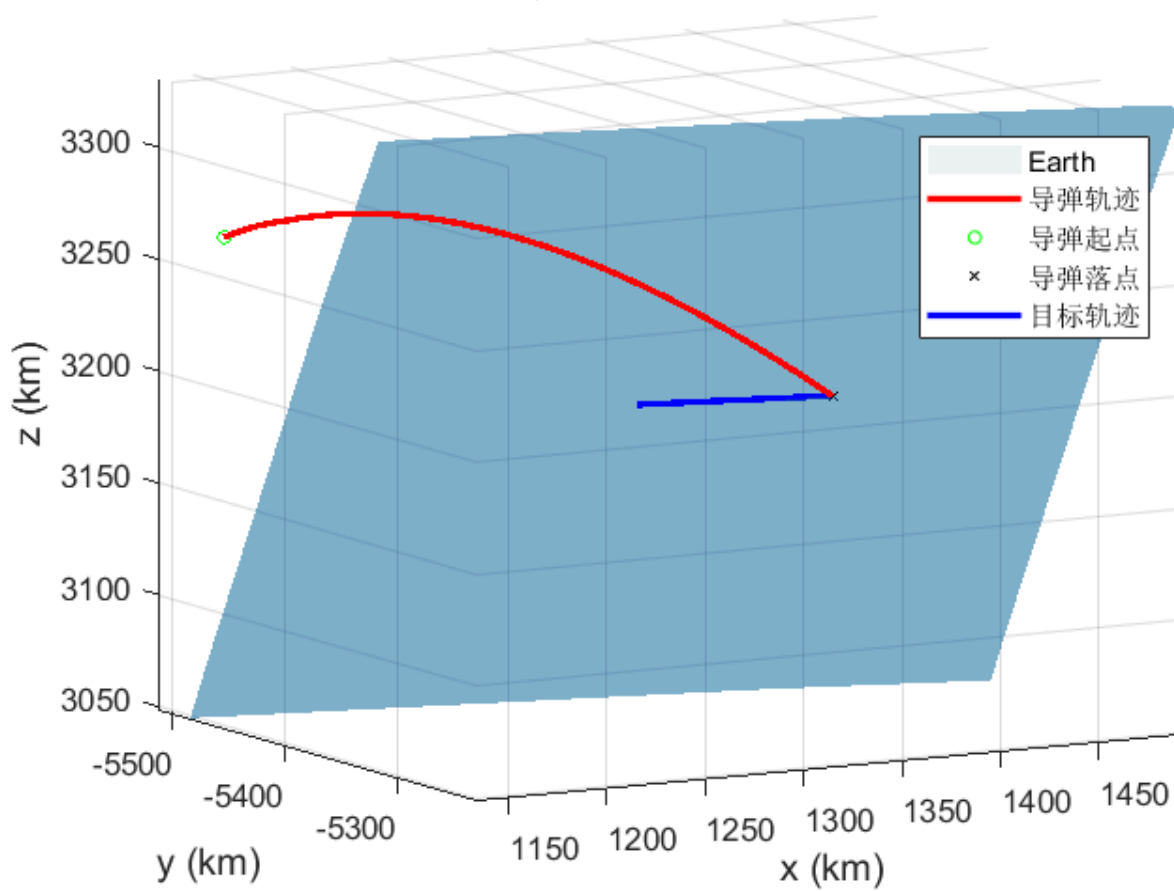
结果分析

导弹飞行时间: 223.185s

脱靶量: 345.3m

关机时间: 26.5s

导弹与目标三维轨迹



导弹与目标间相对距离

