地球科学与环境工程学院

《卫星导航定位原理》

课 间 实 验 报 告 书

设计题目：卫星坐标计算程序

学　　号：2023113911

班　　级：遥感 1 班

姓　　名：于沛豪

日　　期：2025.4.8

目　　录

一、需求分析

1.1 设计内容

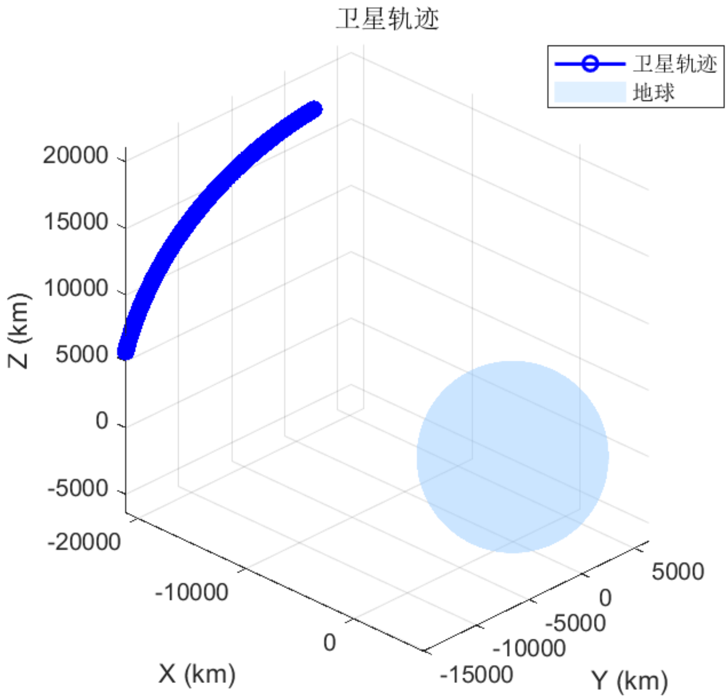


图 1　120 分钟卫星轨迹

1.2 作业需求

[1] 根据星历文件 brdc0010.25p 中的数据，编程计算 GPS 时 2025 年 1 月 1 日 12 点 0 分 00 秒 —20 分 00 秒，每隔 1 分钟的 PRN 1（即 G01）的卫星坐标。

[2] 撰写卫星坐标计算程序设计报告：包括封面、计算原理、程序 代码、解算结果、经验总结等，以.docx 或 PDF 格式提供

[3] 在报告中提供计算结果，列表显示时刻及卫星坐标，也可计算更长的时间段，画图显示

1.2.3 环境需求

Windows11 以及 MATLAB R2024b。

二、实验原理

2.1 坐标计算

（1）求轨道长半轴 a ：

$$a = (\sqrt{a})^2$$

（2）计算平均角速度 n_0 ：

$$n_0 = \sqrt{\frac{GM}{a^3}}$$

式中，GM 为地球引力常数，其值为：GM＝3.9860047×10^14 m^3/s^2 。

（3）观测时刻 t 归化到 GPS 时系的计算公式为：

$$t_k = t - t_{oe}$$

式中：t_k 为相对于参考时刻 t_{oe} 的归化时间，但应计及一个星期（共计 604800 s)的开始或结束。亦即当 $t_k > 302400\mathrm{s}$时，t_k 应减去 604800 s 。当 $t_k < -302400$ s 时，t_k 应加上 604800 s 。

（4）改正平角速度 n ：

$$n = n_0 + \Delta n$$

（5）计算平近点角 M_k ：

$$M_k = M_0 + n \times t_k$$

（6）由已知轨道参数按下式计算偏近点角 E_k ：

$$E_k = M_k + e\sin\ E_k, \left(E_k, M_k \text{ 以弧度 rad 计 }\right)$$
$$E_k = M_k + \left(\frac{180^\circ}{\pi}\right) e\sin\ E_k, \left(E_k, M_k \text{ 以角度计 }\right)$$

上述开普勒方程可用迭代法进行解算，即先令 $E_k = M_k$ ，代人上式。因为 GPS 卫星轨道的偏心率 e 约为 0.01 左右，通常进行迭代计算，便可求得偏近点角 E_k 。

（7）由下两式计算真近点角 f_k ：

$$\left.\begin{aligned}\cos\ f_k &= \frac{(\cos\ E_k - e)}{(1 - e\text{¡Á}\cos\ E_k)}\\ \sin\ f_k &= \frac{\sqrt{1-e^2}\sin\ E_k}{(1 - e\text{¡Á}\cos\ E_k)}\end{aligned}\right\}$$

（8）计算升交距角（或称升交角距）ϕ_k ：

$$\phi_k = f_k + \omega$$

式中：ω 为卫星电文中给出的近地点角距。

（9）计算卫星轨道摄动项改正数：

$$\left.\begin{aligned}\delta u_k &= C_{us}\sin(2\phi_k) + C_{uc}\cos(2\phi_k)\\ \delta r_k &= C_{rs}\sin(2\phi_k) + C_{rc}\cos(2\phi_k)\\ \delta i_k &= C_{is}\sin(2\phi_k) + C_{ic}\cos(2\phi_k)\end{aligned}\right\}$$

（10）计算改正后的向径 r_k ：

$$r_k = a(1 - e\cos\ E_k) + \delta r_k$$

（11）计算改正后的倾角 i_k ：

$$i_k = i_0 + \delta i_k + i\text{¡Á}t_k$$

上式中，i 为轨道倾角变化率，t_k 为相对于参考时刻 t_{oe} 的归化时间。

（12）计算观测瞬间升交点的经度 L_k ：

$$\Omega = \Omega_{t_{oe}} + \dot{\Omega}$$

设本周开始时刻（星期日 0 时）格林尼治恒星时为 $GAST_{\text{week}}$ ，则观测瞬间的格林尼治佰星时为：

$$GAST = GAST_{\text{week}} + \omega_e(t - t_0)$$

式中，ω_e 为地球自转角速度，其值为 $\omega_e = 7.292115\text{x}105\text{rad/s}$; t_0 为一周开始的 GPS 时。有

$$L_k = \text{Ù}_0 + \dot{\text{Ù}}(t - t_{oe}) - \omega_e(t - t_0) = \text{Ù}_0 + \left(\dot{\text{Ù}} - \omega_e\right)t_k - \dot{\text{Ù}}^2 t_{oe}$$

注意：预报星历中给出的 Ù_0 并不是参考时刻 t_{oe} 的升交点赤经 $\Omega_{t_{oe}}$ ，而是该值与本周起始时刻的格林尼治恒星时 GAST week 之差。

（13）计算卫星在轨道平面内的坐标（x_k, y_k, z_k）：

$$\begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = \begin{bmatrix} r_k\cos\ u_k \\ r_k\sin\ u_k \\ 0 \end{bmatrix}$$

（14）最后，计算卫星在协议地球坐标系中的位置（X_k, Y_k, Z_k）：

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix} = R_Z(-L_k)\,R_X(-i_k)\begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix}$$

$$R_Z(-L_k)\,R_X(-i_k) = \begin{bmatrix} cosL_k & -sinL_k cosi_k & sini_k sinL_k \\ sinL_k & cosL_k cosi_k & -cosL_k sini_k \\ 0 & sini_k & cosi_k \end{bmatrix}$$

考虑到地极移动的影响，最后可得到在协议地球坐标系中的空间直角坐标：

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix}_{cts} = R_Y(-X_P)\,R_X(-Y_P)\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix}$$

$$R_Y(-X_P)\,R_X(-Y_P) = \begin{bmatrix} 1 & 0 & X_P \\ 0 & 1 & -Y_P \\ -X_P & Y_P & 1 \end{bmatrix}$$

设两个点 A、B 以及坐标分别为 A(x_1, y_1) B(x_2, y_2)，则 A 和 B 两点之间的距离为:

$$|AB| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{1}$$

2.2 格里高利历至 GPS 时的换算

2.2.1 格里高利历至儒勒日的换算

格里高利历至儒略日的转换公式可表示如下：

$$\text{JD} = \text{INT}[365.25y] + \text{INT}[30.6001(m+1)] + D + h/24 + 1720981.5$$

且有如下说明：若 $M \leqslant 2$，则 $y = Y - 1, m = M + 12$；若 $M > 2$，则 $y = Y, m = M$。

式中：JD 为儒略日；Y为年；M为月；D为日；$h = H + \text{min}/60 + S/3600$, h, min 和 s 分别为时，分和秒； INT[] 为取整函数，有 $\text{INT}[a] \leqslant a$。

2.2.2 儒勒日至 GPS 时的换算

GPS 时的起点是 1980 年 1 月 6 日 0 时，其对应儒略日为 2444244.5 。故儒略日至 GPS 时的转换公式如下：

$$WN = \text{INT}[(\text{JD} - 2444244.5)/7]$$
$$\text{TOW} = \text{mod}\{\text{JD} - 2444244.5, 7\}\text{x}604800.0$$

式中，WN 表示 GPS 时的周数；TOW 表示 GPS 时的周内时间。

三、计算程序

3.1 参数读取

通过 MatLab 编写程序，读取 G01 广播星历参数（参数来自于星历文件 brdc0010.25p），代码如下所示：

```
% 导入广播星历数据（每行含 4 个参数）
ephemeris = [
  4.500000000000E+01,  1.090625000000E+02,  4.528045754138E-09,  3.131427222067E+00;
  5.554407835007E-06,  2.091019414365E-04,  2.112239599228E-06,  5.153755172729E+03;
  3.024000000000E+05,  2.980232238770E-08, -1.783259133478E+00,  2.980232238770E-08;
  9.596575373827E-01,  3.364062500000E+02, -1.305969830383E+00, -8.338561620276E-09;
  2.750114553310E-10,  1.000000000000E+00,  2.347000000000E+03,  0.000000000000E+00
];

% 解包参数
IODE=ephemeris(1,1);Crs = ephemeris(1,2);Delta_n =ephemeris(1,3);M0 = ephemeris(1,4);
Cuc =ephemeris(2,1) e   = ephemeris(2,2);Cus     =ephemeris(2,3);sqrt_a = ephemeris(2,4);
toe =ephemeris(3,1);Cic = ephemeris(3,2);OMEGA   =ephemeris(3,3);Cis    = ephemeris(3,4);
i0  =ephemeris(4,1);Crc = ephemeris(4,2) w       =ephemeris(4,3);OMEGA_DOT = ephemeris(4,4)
;
i_dot = ephemeris(5,1);  % 倾角变化率
```

3.2 时间换算

通过计算原理部分，先将格里高利历转换为儒勒日，后转换为周内秒

```
%% 日期转换：
% 格里高利历 -> 儒略日
jd = fix(365.25 * (year - 1)) + fix(30.6001 * (month + 13)) + day + 12/24 + 17
20981.5;
% 儒略日 -> GPS 周及周内秒
gps_week = fix((jd - 2444244.5) / 7);
tow = mod(jd - 2444244.5, 7) * 86400;  % TOW (周内秒)

% 时间差（卫星钟参考时间修正）
tk0 = tow - toe;
if tk0 > 302400
    tk0 = tk0 - 604800;
elseif tk0 < -302400
    tk0 = tk0 + 604800;
end
tk = ((0:point_num) * 60) + tk0;  % 从 tk0 开始每分钟 1 个样本
```

3.3 常数定义

定义所需常数（书中地球引力常数编写有误，见经验总结部分）

```
%% 常数
GM = 3.9860047e14;        % 地心引力常数 (m^3/s^2)
We = 7.2921151467e-5;     % 地球自转角速度 (rad/s)
a = sqrt_a^2;             % 轨道长半轴
n0 = sqrt(GM / a^3);      % 平均角速度
n = n0 + Delta_n;         % 改正后角速度
```

3.4 轨道参数计算

通过迭代法解 Kepler 方程，为保持收敛精度，此处规定 1e-9。在真近点角计算时采用 atan2 函数，不用区分象限。注意摄动改正采用点乘而非乘。

```
%% 轨道参数计算
Mk = M0 + n * tk;         % 平近点角
Ek = Mk;                  % 初始化偏近点角

% 解 Kepler 方程（迭代法）
for iter = 1:10000
    Ek_next = Mk + e .* sin(Ek);
    if all(abs(Ek_next - Ek) < 1e-9)
        disp('偏近点角达到收敛');
        break;
    end
    Ek = Ek_next;
end

% 真近点角计算
fk = atan2(sqrt(1 - e^2) .* sin(Ek), cos(Ek) - e);

% 升交角距
phi_k = fk + w;

% 摄动改正
delta_u = Cus .* sin(2 * phi_k) + Cuc .* cos(2 * phi_k);
delta_r = Crs .* sin(2 * phi_k) + Crc .* cos(2 * phi_k);
delta_i = Cis .* sin(2 * phi_k) + Cic .* cos(2 * phi_k);

% 改正后参数
rk = a * (1 - e .* cos(Ek)) + delta_r;          % 距离
ik = i0 + i_dot .* tk + delta_i;                % 倾角
uk = phi_k + delta_u;                           % 升交距角
```

3.5 卫星坐标计算

使用变量 point_num 作为想要计算的点的数量（因为 matlab 的括号取值同 python 本身 array，均为左闭右开，故此处自加一）

```
%% 卫星坐标计算
point_num=point_num+1;
XYZ = zeros(point_num, 3);  % 初始化

for i = 1:point_num
    % 卫星轨道平面坐标
    x_orb = rk(i) * cos(uk(i));
    y_orb = rk(i) * sin(uk(i));

    % 轨道平面 -> ECEF 转换矩阵
R = [cos(OMEGA_k(i)),sin(OMEGA_k(i))*cos(ik(i)),sin(OMEGA_k(i))*sin(ik(i));
     sin(OMEGA_k(i)),cos(OMEGA_k(i))*cos(ik(i)),cos(OMEGA_k(i)) * sin(ik(i));
         0          ,         sin(ik(i))                ,cos(ik(i))];

    % 坐标变换
    XYZ(i,:) = (R * [x_orb; y_orb; 0])';
end
```

3.6 输出与可视化绘图

通过 plot3 绘制三维图像，除以 1000 使坐标轴化为 km 为单位，控制视窗大小。地球的绘制假想为一个 6371km 为半径的球体。

```
disp(XYZ);


%% 可视化轨迹
figure;
plot3(XYZ(:,1)/1e3, XYZ(:,2)/1e3, XYZ(:,3)/1e3, 'b-o', 'LineWidth', 1.5);
grid on;
xlabel('X (km)');
ylabel('Y (km)');
zlabel('Z (km)');
title('卫星轨迹');
axis equal;
view(45, 30);

% 绘制地球
[xe, ye, ze] = sphere(50);
earth_radius = 6371;
hold on;
```

```
surf(earth_radius*xe, earth_radius*ye, earth_radius*ze, ...
    'FaceColor', [0.6 0.8 1], 'EdgeColor', 'none', 'FaceAlpha', 0.3);
legend('卫星轨迹', '地球');
```

四、解算结果

4.1 解算后坐标

表 1 解算所得 XYZ 坐标

分钟	X	Y	Z
0	-16009488.1292602	-2472417.97476465	21056044.4043865
1	-16048742.1192336	-2627751.77382215	21007295.0541760
2	-16088432.4469482	-2782589.40924493	20956938.0116368
3	-16128552.4365917	-2936918.20799287	20904977.1286531
4	-16169095.2770838	-3090725.58518088	20851416.3797379
5	-16210054.0232472	-3243999.04597745	20796259.8617306
6	-16251421.5970013	-3396726.18748700	20739511.7934857
7	-16293190.7885812	-3548894.70061707	20681176.5155521
8	-16335354.2577794	-3700492.37192893	20621258.4898429
9	-16377904.5352118	-3851507.08547185	20559762.2992966
10	-16420834.0236072	-4001926.82460112	20496692.6475285
11	-16464134.9991196	-4151739.67377821	20432054.3584731
12	-16507799.6126634	-4300933.82035453	20365852.3760175
13	-16551819.8912718	-4449497.55633666	20298091.7636256
14	-16596187.7394774	-4597419.28013444	20228777.7039530
15	-16640894.9407147	-4744687.49829012	20157915.4984532
16	-16685933.1587455	-4891290.82718967	20085510.5669747
17	-16731293.9391051	-5037217.99475475	20011568.4473486
18	-16776968.7105710	-5182457.84211565	19936094.7949677
19	-16822948.7866517	-5326999.32526542	19859095.3823570
20	-16869225.3670979	-5470831.51669334	19780576.0987341

4.2 可视化绘图

因为所给星历文件 brdc0010.25p 中 G01 数据为每 2 小时发布一次广播星历，故可视化绘图采用 120min 画图显示，若时间段过短，将由于数据过少绘图效果不会太好，若数据过长则由于星历文件参数过时则造成误差过大，故绘图如下所示：

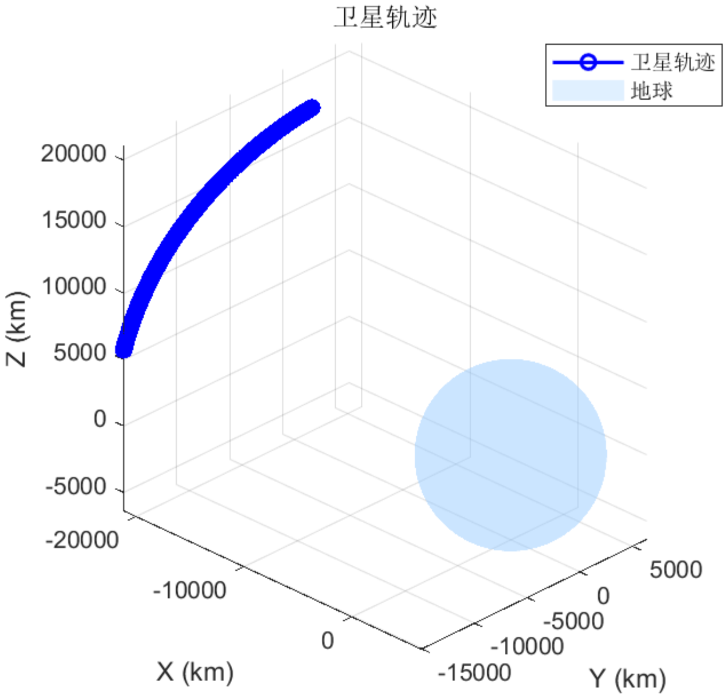


图 2　120 分钟卫星轨迹

五、经验总结

5.1 小插曲

在《卫星导航定位原理》这本书（ISBN：978-7-307-14817-5），版次 2015 年 1 月第一版，2023 年 2 月第 5 次印刷。于第 89 页（4.60）公式下方 GM 的值有错。书中数据为 GM＝3.9860047 × 1014 m^3/s^2。实际应为 GM＝3.9860047 × 10^{14} m^3/s^2

已发邮件到武汉大学出版社。

发现贵社出版的书的一点小笔误

发起会议

2025-04-08 22:02:43

发件人："于沛豪"<8393455@my.swjtu.edu.cn>

收件人：cbs22@whu.edu.cn

附　件：1 个(IMG20250408220112.jpg) 查看附件

▶ 发送成功 查看详情 共发给 1 个收件人，其中1 个 成功到达对方服务器

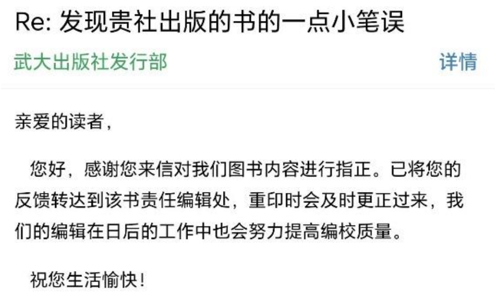
我是一名遥感专业学生，目前正在学习《卫星导航定位原理》这本书（ISBN：978-7-307-14817-5），版次2015年1月第一版，2023年2月第5次印刷。于第89页（4.60）公式下方GM的值是否应为...x10的14次方，而非x1014，见附件图片。

普通附件(1 个)

IMG20250408220112.jpg (237.01K)

下载　预览　保存到个人网盘

得到回应如下：



Re: 发现贵社出版的书的一点小笔误

武大出版社发行部　　详情

亲爱的读者，

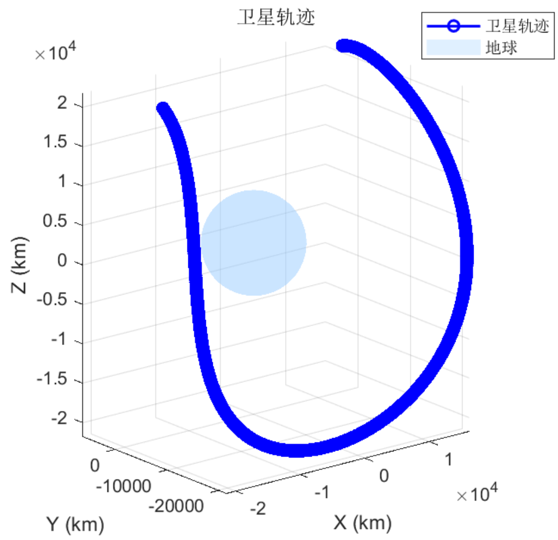
您好，感谢您来信对我们图书内容进行指正。已将您的反馈转达到该书责任编辑处，重印时会及时更正过来，我们的编辑在日后的工作中也会努力提高编校质量。

祝您生活愉快！

5.2 反思与改进

原程序只支持基于一个星历文件的预测，若数据过长则由于星历文件参数过时则造成误差过大。则可以对转换函数整体作为 for 循环的内容，以星历文件个数（两小时一次）为循环，便可实现将不同星历文件分别预测 120min 卫星位置后叠加到变量 Total_XYZ 中，只需一点小改动便可使预测长时间数据更加精准，因为若预测过长时间段则会星历文件参数过时则造成误差过大，该种办法相当于每个星历文件预测 120 点后叠加。

通过预测更长时间段（2025-1-1 12:00:00 - 2025-1-2 00:00:00）所生成的图像如下：



因为 ECEf 是地心地固坐标系，是转动的，所以无法呈现出标准椭圆。至于更好显示出围绕地球转动的效果则可以乘以旋转矩阵改变参数。

5.3 总结

通过这次实验，我加深了对 Matlab 的理解，其中我明白了规范的代码编写方法，包括如何编写变量名，atan 与 atan2 的细微区别，format 的设置，收敛精度的确定，以及 3d 绘图，即 plot3 的用法。这些都为我以后的学习帮助了许多。

从查文档了解各个参数的意义，到完成程序编写后，直到最后输出图像的时候，那一刻的自豪感真是无可比拟。

程序已随实验报告打包提交。