Этап 2 «Моделирование»

На предыдущем этапе получено решение навигационной задачи с помощью программы вторичной обработки измерений RTKLIB. В процессе работы она рассчитывает положение спутников на соответствующий момент сигнального времени. При этом используются эфемериды - параметры некоторой модели движения спутника. В разных ГНСС эти модели разные, а значит отличается и формат эфемерид, и алгоритмы расчета положения спутника.

Требуется реализовать на языке Matlab функцию расчета положения спутника GPS на заданный момент по шкале GPST. Построить трехмерные графики множества положений спутника GPS с системным номером. Построить SkyView за указанный временной интервал.

Таблица эфемерид:

|  |  |
| --- | --- |
| toe, сек | 288000 |
| A , м | 26559820.385 |
| e | 0.00555263 |
| i0, град | 54.40784 |
| OMEGA0, град | -4.78780 |
| omega ,град/с | 40.89810 |
| M0, град | -154.32800 |
| deltan, град/с | 2.9120\*10^-7 |
| OMEGAdot, рад/с | -4.69290\*10^-7 |
| IDOT,град/с | 1.6494\*10^-8 |
| cuc, рад | -8.0094\*10^-7 |
| cus, рад | 7.4618\*10^-6 |
| crc, м | 2.3034\*10^2 |
| crs, м | -1.2\*10^1 |
| cic, рад | 3.3528\*10^-8 |
| cis, рад | -1.3970\*10^-7 |

Листинг программы:

mu = 3.986005\*10^14;

dOmega = 7.2921151467\*10^-5;

TOW = 302418;

t0e = 309600+18

i0 = 54.40784\*(pi/180); A = 26559820.385;

e = 0.00555263;

omegadot = -4.69290\*10^-7 \*(pi/180);

omega0 = -4.78780\*(pi/180);

omega = 40.89810\*(pi/180);

M0 = -154.32800\*(pi/180);

deltan = 2.9120\*10^-7 \*(pi/180);

IDOT = 1.6494\*10^-8 \*(pi/180);

Cuc = -8.0094\*10^-7;

Cus = 7.4618\*10^-6;

Crc = 2.3034\*10^2;

Crs = -1.2\*10^1;

Cic = 3.3528\*10^-8;

Cis = -1.3970\*10^-7;

shirota = 55.45;

dolgota = 37.42;

visota = 500;

for i = 1:900

TOW = TOW + i;

n0 = sqrt(mu/(A^3));

Tk = TOW - t0e;

Mk = M0 + (n0 + deltan)\*Tk;

E = 0;

while(true)

Ek = Mk + e\*sin(E);

if (abs(E - Ek) <= 10^-8)

break;

end

E = Ek;

end

nu = atan2(((sqrt(1-e^2)\*sin(Ek))/(1 - e\*cos(Ek))), ((cos(Ek) - e)/(1 - e\*cos(Ek))));

fi = nu + omega;

deltaU = Cus\*sin(2\*fi) + Cuc\*cos(2\*fi);

deltaR = Crs\*sin(2\*fi) + Crc\*cos(2\*fi);

deltaI = Cis\*sin(2\*fi) + Cic\*cos(2\*fi);

U = fi + deltaU;

R\_Zeml = A\*(1 - e\*cos(Ek)) + deltaR;

I = i0 + deltaI + IDOT\*Tk;

Xorb = R\_Zeml\*cos(U);

Yorb = R\_Zeml\*sin(U);

OmegaK = omega0 + (omegadot - dOmega)\*Tk - dOmega\*t0e;

x = Xorb\*cos(OmegaK) - Yorb\*cos(I)\*sin(OmegaK);

y = Xorb\*sin(OmegaK) + Yorb\*cos(I)\*cos(OmegaK);

z = Yorb\*sin(I);

R(i,:)=[x y z];

teta = dOmega\*Tk;

xeci = x\*cos(teta) - y\*sin(teta);

yeci = x\*sin(teta) + y\*cos(teta);

zeci = z;

ECI(i,:) = [xeci yeci zeci];

[East, North, Up] = ecef2enu(x, y, z, shirota, dolgota,visota, wgs84Ellipsoid);

SAT = sqrt(East^2 + North^2 + Up^2);

AZ(i) = atan2(East, North);

EL(i) = -asin(Up/SAT)\*180/pi + 90;

end

[X, Y, Z] = sphere(10);

figure(1);

plot3(R(:,1),R(:,2),R(:,3))

hold on;

grid on;

surf(X\*6371\*10^3, Y\*6371\*10^3, Z\*6371\*10^3);

figure(2);

plot3(ECI(:,1),ECI(:,2),ECI(:,3));

hold on;

grid on;

surf(X\*6371\*10^3, Y\*6371\*10^3, Z\*6371\*10^3);

figure(3);

polar (AZ, EL);

hold on;

grid on;

camroll(90);

Полученные графики:

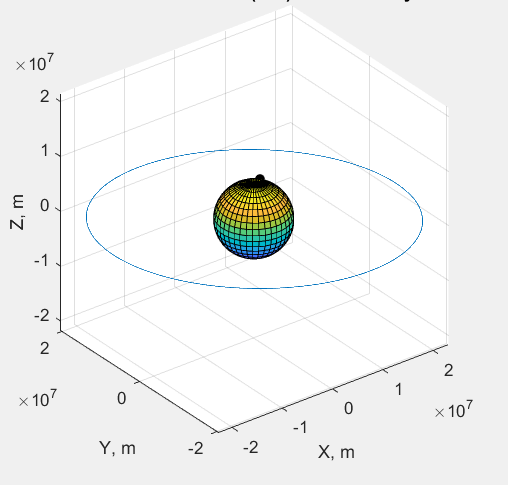


Рис.1 траектория в ECI

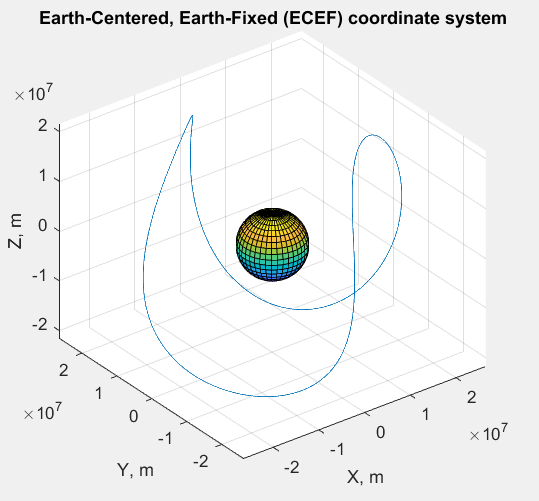


Рис.2 траектория в ECEF

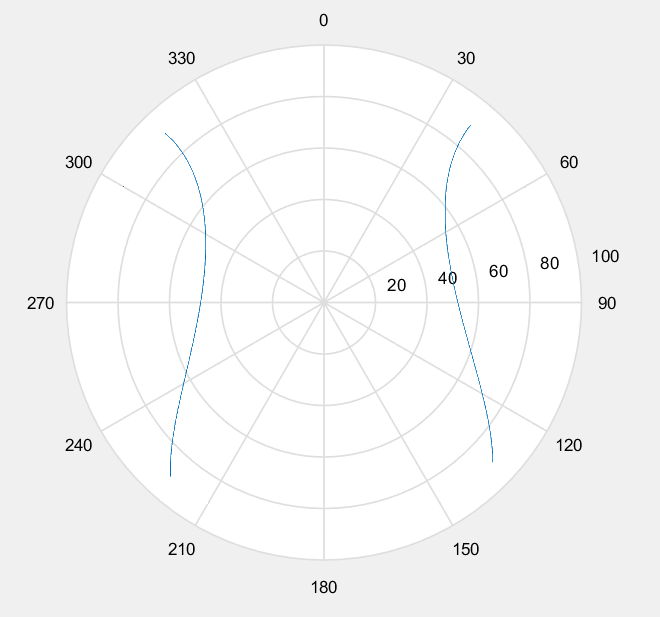


Рис.4 SkyView