**Национальный исследовательский университет**

**«МЭИ»**

**Институт радиотехники и электроники**

**Кафедра радиотехнических систем**

**Аппаратура потребителей СРНС**

Курсовой проект

Группа: ЭР-15-14

ФИО студента: Опанасенко А.А.

ФИО преподавателя: Корогодин И.В.

**Москва, 2019г**

**Постановка задачи**

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.

Одна из самых простых и удобных моделей - [в системе GPS](http://navipedia.org/index.php/GPS_and_Galileo_Satellite_Coordinates_Computation).

Требуется реализовать на языке Matlab или Python функцию расчета положения спутника GPS на заданный момент по шкале GPST. В качестве эфемерид использовать данные, полученные на предыдущем этапе.

Построить трехмерные графики множества положений спутника GPS с системным номером, соответствующим номеру студента по списку. Графики в двух вариантах: в СК ECEF WGS84 и соответствующей ей инерциальной СК. Положения должны соответствовать временному интервалу с 12:00 13.02.19 до 00:00 14.02.19. Допускается использовать одни и те же эфемериды на весь рассматриваемый интервал.

Построить SkyView за указанный временной интервал (напоминаю, антенна на крыше корпуса Е) и сравнить результат с [Trimble GNSS Planning Online](http://www.trimble.com/gnssplanningonline/), полученный на прошлом этапе.

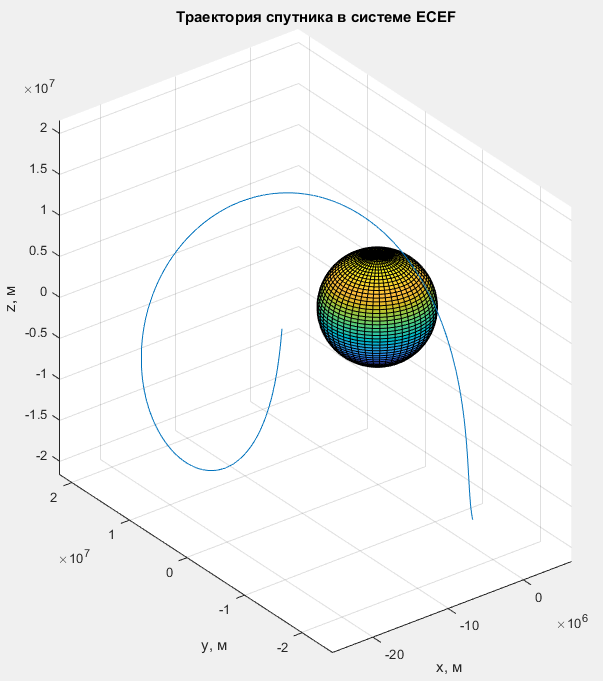
**Решение**

Алгоритм нахождения положения спутника и построение трехмерных графиков множества положений спутника GPS реализован в Matlab.

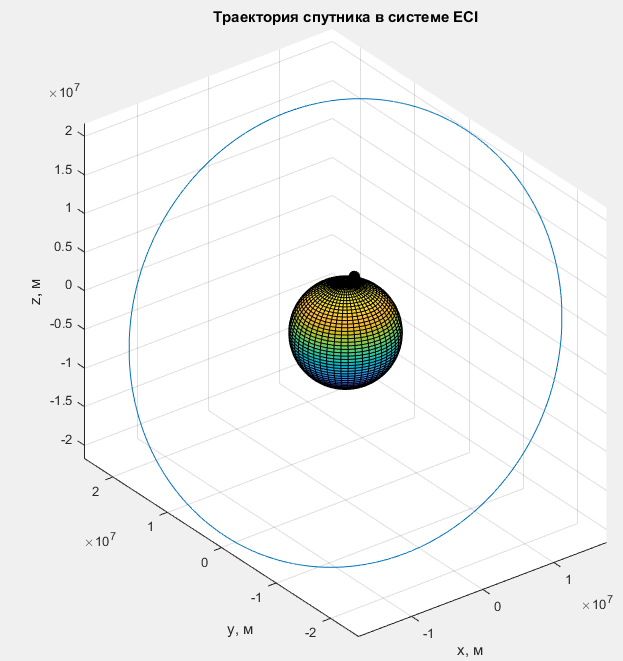
Положение спутника на момент времени 13/02/2019 12:00:00:

Расстояние рассчитано с учетом радиуса Земли. Таким образом спутник находится на высоте 20182 км, что близко к среднему значению орбиты спутников группировки GPS в 20800 км.

Построим трехмерные графики множества положений спутника GPS с системным номером 9. Графики изобразим в двух вариантах: в системе координат ECEF и в системе координат ECI.

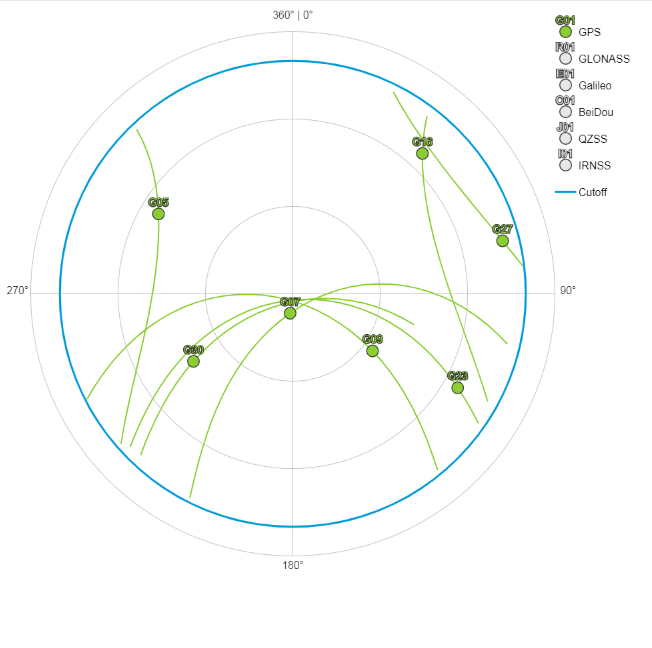


***Рис. 1. Траектория спутника GPS №9 в системе координат ECEF.***

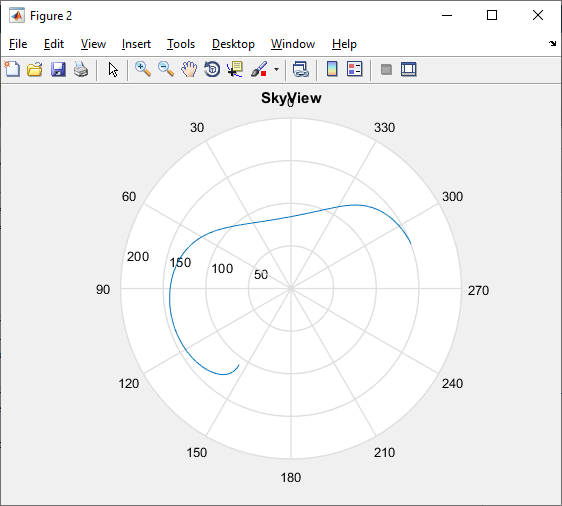


***Рис. 2. Траектория спутника GPS №9 в системе координат ECI.***

Построим SkyView за указанный временной интервал (напоминаю, антенна на крыше корпуса Е) и сравнить результат с [Trimble GNSS Planning Online](http://www.trimble.com/gnssplanningonline/), полученный на прошлом этапе.



***Рис. 3. SkyView, полученная на прошлом этапе.***



***Рис. 4. SkyView, полученная по координатам спутника.***

**Листинг программы**

%{

Курсовой проект. Часть 2.

Студент: Опанасенко А.А,

Группа: ЭР-15-14

%}

clear all;

clc;

% Исходные данные

% Square root of semi-major axis

A = 26560971.874;

% Ephemerides reference epoch in seconds within the week

Toe = 288000;

% Mean anomaly at reference epoch

M0 = degtorad(2.01704);

% Longitude of ascending node at the beginning of the week

omega\_zero = degtorad(55.82613);

% Argument of perigee

omega = degtorad(98.53938);

% Rate of node's right ascension

omega\_dot = degtorad(-4.6497e-7);

% WGS 84 value of earth's rotation rate

omega\_dot\_e = 7.2921151467e-5;

% Eccentricity

e = 0.00146475;

% Inclination at reference epoch

I0 = degtorad(54.57146);

% Mean motion difference

delta\_n = degtorad(2.7403e-7);

M = 3.986005\*10^14;

% Rate of inclination angle

IDOT = degtorad(-1.8028e-8);

Cus = 8.6408e-6;

Cuc = -8.6054e-7;

Crs = -1.5281e1;

Crc = 2.0794e2;

Cis = -4.2841e-8;

Cic = -8.7544e-8;

% Решение

for k=1:43200

T = 302400 + k;

Tk = T - Toe;

% Time from ephemeris reference epoch

if (Tk > 302400)

Tk = Tk - 604800;

elseif (Tk < -302400)

Tk = Tk + 604800;

end

% Compute mean motion

n0 = sqrt(M/A^3);

% Correct mean motion

n = n0 + delta\_n;

% Mean anomaly

Mk = M0 + n\*Tk;

Ek\_prev = 0;

while(true)

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

if (abs(Ek\_prev - Ek) <= 0.0000001)

break;

end

Ek\_prev = Ek;

end

% True Anomaly

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

% Argument of Latitude

Fk = Vk + omega;

%Second Harmonic Perturbations

delta\_Uk = Cus\*sin(2\*Fk) + Cuc\*cos(2\*Fk);

delta\_Rk = Crs\*sin(2\*Fk) + Crc\*cos(2\*Fk);

delta\_Ik = Cis\*sin(2\*Fk) + Cic\*cos(2\*Fk);

% Correct Argument of Latitude

Uk = Fk + delta\_Uk;

% Correct Radius

Rk = A\*(1 - e\*cos(Ek)) + delta\_Rk;

% Correct Inclination

Ik = I0 + delta\_Ik + IDOT\*Tk;

% Correct longitude of ascending node

Wk = omega\_zero + (omega\_dot - omega\_dot\_e)\*Tk - omega\_dot\_e\*Toe;

% Positions in orbitalplane

xk = Rk\*cos(Uk);

yk = Rk\*sin(Uk);

% Earth-fixed coordinates

% Координаты в системе ECEF

xk\_fixed(k) = xk\*cos(Wk) - yk\*cos(Ik)\*sin(Wk);

yk\_fixed(k) = xk\*sin(Wk) + yk\*cos(Ik)\*cos(Wk);

zk\_fixed(k) = yk\*sin(Ik);

rangeEcef(k) = sqrt((xk\_fixed(k))^2 + (yk\_fixed(k))^2 + (zk\_fixed(k))^2);

% Переведем координаты в систему ECI

theta = omega\_dot\_e\*Tk;

xk\_eci(k) = xk\_fixed(k)\*cos(theta) - yk\_fixed(k)\*sin(theta);

yk\_eci(k) = xk\_fixed(k)\*sin(theta) + yk\_fixed(k)\*cos(theta);

zk\_eci(k) = zk\_fixed(k);

rangeEci(k) = sqrt((xk\_eci(k))^2 + (yk\_eci(k))^2 + (zk\_eci(k))^2);

% Построим диаграмму SkyView

moscowLatitude = 55.75;

moscowLongitude = 37.62;

moscowHeight = 150;

[East, North, Up] = ecef2enu(xk\_fixed(k), yk\_fixed(k), zk\_fixed(k), moscowLatitude, moscowLongitude, moscowHeight, wgs84Ellipsoid);

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

elevation(k) = -asin(Up/rangeFromRecieverToSatellite)\*180/pi + 90;

azimuth(k) = atan2(East, North);

end

% Построение SkyView

figure;

polar (azimuth, elevation);

title('SkyView');

grid on;

camroll(90);

% Графически отобразим траекторию спутника №9 в системе ECI и ECEF

thetavec = linspace(0, pi, 50);

phivec = linspace(0, 2\*pi, 50);

[th, ph] = meshgrid(thetavec,phivec);

R = 6.371\*10^6;

x = R.\*sin(th).\*cos(ph);

y = R.\*sin(th).\*sin(ph);

z = R.\*cos(th);

latitude = 55\*pi/180;

longitude = 37\*pi/180;

coordMoscowX = R\*cos(latitude)\*cos(longitude);

coordMoscowY = R\*cos(latitude)\*sin(longitude);

coordMoscowZ = R\*sin(latitude);

% В системе ECEF

figure;

surf(x, y, z);

axis equal;

hold on;

plot3(xk\_fixed(1,:), yk\_fixed(1,:), zk\_fixed(1,:));

axis vis3d;

grid on;

title('Траектория спутника в системе ECEF');

xlabel('x, м');

ylabel('y, м');

zlabel('z, м');

% В системе ECI

figure;

surf(x, y, z);

axis equal;

hold on;

plot3(xk\_eci(1,:), yk\_eci(1,:), zk\_eci(1,:), coordMoscowX, coordMoscowY, coordMoscowZ, 'k.','MarkerSize', 30);

axis vis3d;

grid on;

title('Траектория спутника в системе ECI');

xlabel('x, м');

ylabel('y, м');

zlabel('z, м');