# Homework 6

Date of Submission:
March 1, 2019

---

Thorne Wolfenbarger
wolfent1@my.erau.edu

Submitted to:
Professor Kaela Martin
College of Engineering

In Partial Fulfillment
of the Requirements of

AE 313
Space Mechanics
Spring, 2019

# AE 313 Homework 5

1. When $\theta_2^* = 280°$, find current $\theta_{ERA}$ and the latitude and longitude.
$\theta_{ERA} = 30.58$
$latitude\ \phi = 25.4745°$
$longitude\ \lambda = 76.3665°$

2. For a Prescott ground station what is the elevation and azimuth relative to the ground station? Can the ground station "see" the spacecraft?
$\theta_E = -55.4871°$
$\theta_A = -170.8419°$
The ground station can not "see" the spacecraft due to the fact that $\theta_E$ is negative. This means that it is below the visible horizon.

3. Use MATLAB to produce the ground track using "geoshow" for the time between the two positions. Confirm the initial and final latitude and longitude.

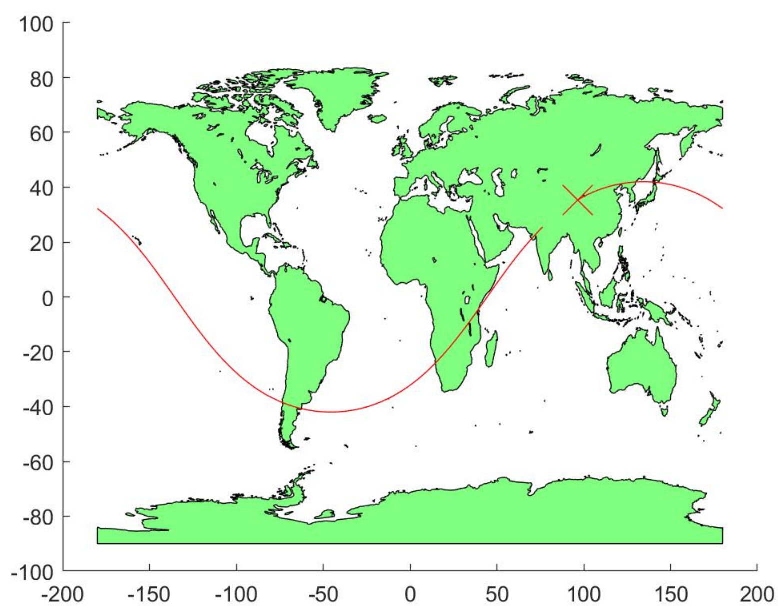


Figure 1: MATLAB Geoshow Plot

Initial latitude and longitude:
$\phi = 35.4143°, \lambda = 96.5763°$
Final latitude and longitude:
$\phi = 25.4745°, \lambda = 76.3665°$

4. Use GMAT to produce a ground track of the orbit to confirm the MATLAB code.

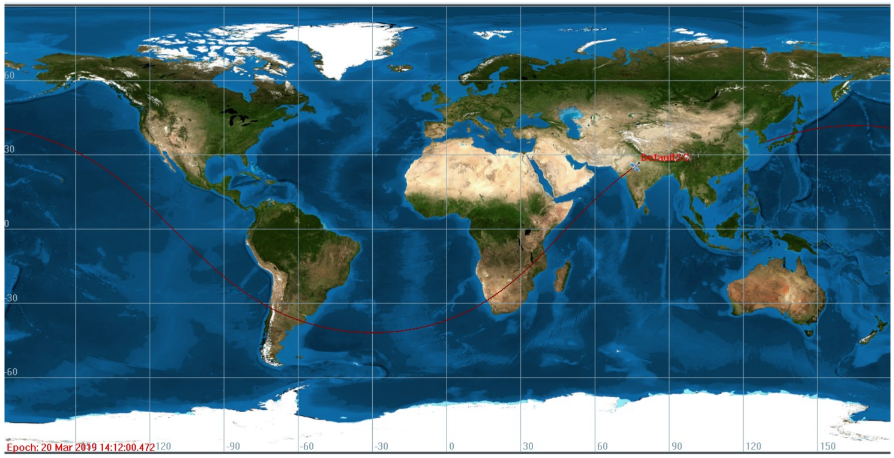


Figure 2: GMAT Spacecraft Plot

$\vec{r}_1^{ECI} = -2163.29\hat{x} + 7099.49\hat{y} + 3535.94\hat{z}\ km$
$\vec{v}_1^{ECI} = -4.72\hat{x} - 4.52\hat{y} + 3.05\hat{z}\ \frac{km}{s}$
GMAT Initial latitude and longitude:
$\phi = 35.50°, \lambda = 129.73°$
GMAT Final latitude and longitude:
$\phi = 25.56°, \lambda = 76.41°$
The initial latitude is within 0.2° but the initial longitude is not. This is unexpected becuase the GMAT final latitudes and longitudes are both within the 0.2° limit. This difference in initial longitudes is attributed to the fact that the matlab code did not take the rotation of the earth into account. This makes sense because the differe is only seen in the resulting longitude. To fix this error I would need to recalculate the Tu value at each new successive $\theta^*$

HW6.m

```
clc; close all; clear all

MU = 398600;
a = 8861; % km
AOP = 120; % deg
RAAN = 75; % deg
e = 0.2;
inc = 42; % deg
AOL1 = 60; % deg

true_a1 = AOL1-AOP; %deg
true_a2 = 280; % deg

syms E
eq = tand(true_a1/2) == sqrt((1+e)/(1-e))*tan(E/2);
E1 = double(solve(eq,E));
eq = tand(true_a2/2) == sqrt((1+e)/(1-e))*tan(E/2);
E2 = double(solve(eq,E));

syms t1 t2
eq = sqrt(MU/a^3)*t1 == E1-e*sin(E1);
t1 = double(solve(eq,t1));
eq = sqrt(MU/a^3)*t2 == E2-e*sin(E2);
t2 = double(solve(eq,t2));

syms dt
period = 2*pi*sqrt(a^3/MU);
dt = t2 - t1 + period;
dt_days = dt/60/60/24;

syms theta_era
%
tu = juliandate(datetime('2019-03-20 12:00:00')) - 2451545 + dt_days;
theta_era = 2*pi*(0.7790572732640 + 1.00273781191135448*tu);
theta_era = mod(theta_era,2*pi)*180/pi;


syms lat
eq = sind(lat) == sind(inc)*sind(AOP+true_a2);
lat = double(solve(eq, lat));
lat = lat(1); %Lat is only defined from -90 to 90

syms long
```

```
alpha = atan2d(cosd(inc)*sind(AOP+true_a2),cosd(AOP+true_a2));
eq = long == alpha + RAAN - theta_era;
long = double(solve(eq, long)); % Long will always be correct

syms r2
lat_gs = 34.54;
long_gs = -112.4685;
z_gs = 1.62;
AOL2 = AOP + true_a2;
p = a*(1-e^2);
r2 = p/(1+e*cosd(true_a2));
vr2 = [r2 0 0]';
rth_eci = rot_rth_eci(RAAN, inc, AOL2);
eci_ecef = rot_eci_ecef(theta_era);
ecef_sez = rot_ecef_sez(long_gs, lat_gs);
vr2_sez = ecef_sez*eci_ecef*rth_eci*vr2;
vr_e_gs = [0 0 z_gs+6378]';
r_gs_sc = vr2_sez - vr_e_gs;

azimuth = atan2d(r_gs_sc(2),r_gs_sc(1));
elevation = asind(r_gs_sc(3)/norm(r_gs_sc));

% Problem 4
true_a_array = linspace(true_a1, true_a2, 1000);
theta_array = AOP + true_a_array;
lat_array = asind(sind(inc)*sind(theta_array));

alpha_array = atan2d(cosd(inc)*sind(theta_array),cosd(theta_array));
long_array = alpha_array + RAAN - theta_era;

figure; hold on;
geoshow("landareas.shp","FaceColor",[0.5 1.0 0.5])
geoshow(lat_array, long_array,'Color','red')
geoshow(lat_array(1),long_array(1),'DisplayType','Point','Marker','x','Markersize',20)

% GMAT LAT LONG
% 25.55875554722227 76.41433137574843



function A = rot_rth_eci(o,i,t)

% o : Omega, Longitude of the Ascending Node (RAAN)
```

```
% i : i, inclination
% t : theta, Argument of Latitude

A = [cosd(o)*cosd(t)-sind(o)*cosd(i)*sind(t), -cosd(o)*sind(t)-sind(o)*cosd(i)*cosd(t), sind(o)*sind(i); ...
        sind(o)*cosd(t)+cosd(o)*cosd(i)*sind(t), -sind(o)*sind(t)+cosd(o)*cosd(i)*cosd(t), -cosd(o)*sind(i); ...
        sind(i)*sind(t), sind(i)*cosd(t), cosd(i)];

end

function A = rot_eci_ecef(t)

% t : theta_era, Earth rotation angle

A = [cosd(t), sind(t), 0; -sind(t), cosd(t), 0; 0, 0, 1];
end

function A = rot_ecef_sez(l,p)

% l : lamda_gs, longitude of the ground station
% p : phi_gs, latitude of the ground station

A = [sind(p)*cosd(l), sind(p)*sind(l), -cosd(p);
    -sind(l), cosd(l), 0;
    cosd(p)*cosd(l), cosd(p)*sind(l), sind(p)];

end
```