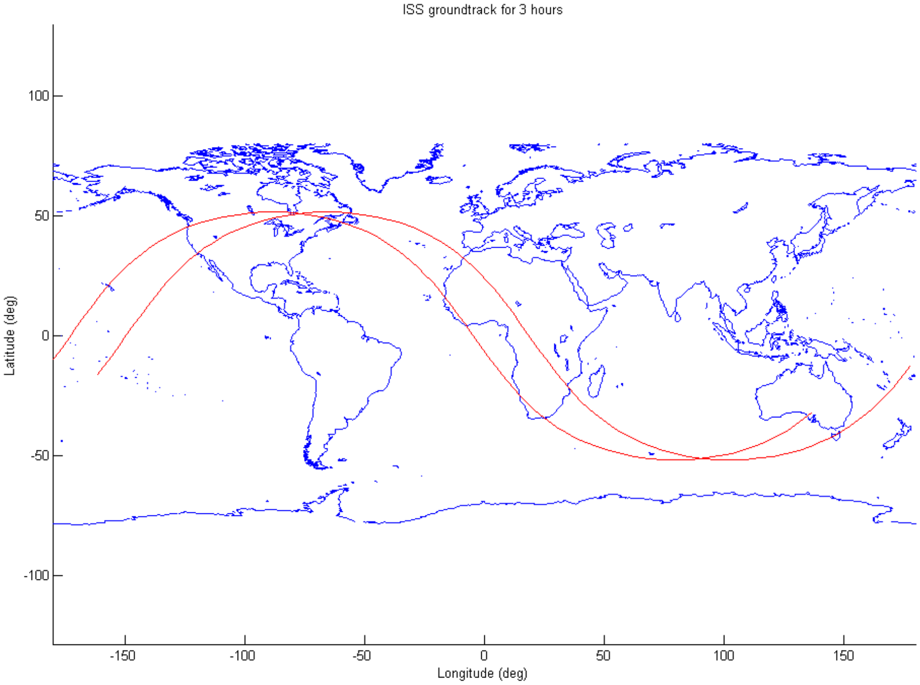
HW6 Problem 1

```
fprintf('\n');
clearvars -except function_list hw_pub toolsPath
close all
CelestialConstants; % import useful constants

% Oct 10, 2014
hr = .86929405* 24;
minute = (hr - floor(hr))*60;
second = (minute - floor(minute))*60;
MDT_offset = -6;
% The epoch is 10-10-2014 floor(hr):floor(min):sec GMT
fprintf(['Epoch is 10-10-2014 ' num2str(floor(hr)) ':' ...
    num2str(floor(minute)) ':' num2str((second)) ' GMT\n']);
fprintf(['       = 10-10-2014 ' num2str(floor(hr)+MDT_offset) ':' ...
    num2str(floor(minute)) ':' num2str((second)) ' MDT\n']);
fprintf('\t(used www.epochconverter.com to find day from day-of-year\n');
time_to_init = 19-(hr+MDT_offset);
fprintf(['Must propagate ' num2str(time_to_init) ' hours (' ...
    num2str(time_to_init/24) ' days) to reach \n\t10-10-2014 19:00:00 MDT\n']);

load('Map.dat');

i = 051.6467 * pi/180;
RAAN = 246.1653 * pi/180;
e = 0.0002545;
w = 242.3213 * pi/180;
M = 215.0173 * pi/180;
n = 15.50348149909277 * 2*pi /24/3600; % rad/s
a = (Earth.mu/n/n)^(1/3);

% Get Greenwich Mean Siderial Time for the beginning of the scenario
JD = computeJD(2014, 10, 11, 1, 0, 0);

dt = 60; % sec
prop_time = 3*3600; %sec
time_vec = 0:dt:prop_time;
num_pts = length(time_vec);
geocen_lat_vec= zeros(length(time_vec),1);
lat_vec = zeros(length(time_vec),1);
lon_vec = zeros(length(time_vec),1);
r_ecef_store = zeros(3,length(time_vec));
cnt = 0;

% First, propagate to the init time.
M = M + n*time_to_init*3600;
while M > 2*pi %Assumes dt < Period of orbit...
    M = M - 2*pi;
end

% Now propagate for the prescribed time.
```

```
for t = time_vec
    cnt = cnt + 1;
    if t > 0
        M = M + n*dt;
        % Increment date to find GMT angle
        JD = JD + dt/86400;
    end
    if M > 2*pi %Assumes dt < Period of orbit...
        M = M - 2*pi;
    end

    % Get true anom
    E = M2E(M,e);
    f = E2f(E,e);

    % Cartesian ECI state
    [r,v] = OE2cart(a,e,i,RAAN,w,f,Earth.mu);

    % Find the Greenwich solar angle
    greenwich_time = computeLocalSiderealTime(JD,0,Earth.spin_rate);

    % Coords in ECEF
    r_ecef = eci2ecef(r, greenwich_time);
    % Save it for Problem 2
    r_ecef_store(:,cnt) = r_ecef;

    % Resultant lat(geocentric), lon, altitude
    [lat, lon, alt] = ECEF2latlonalt(r_ecef);

    % Record the result for the groundtrack plot
    geocen_lat_vec(cnt) = lat;
    lat_vec(cnt) = atan2(tan(lat),(1-Earth.flattening*Earth.flattening)); %FIXME!!
    lon_vec(cnt) = lon;
end

discontinuity_idx = [];
for ii = 2:length(lon_vec)
    if abs(lon_vec(ii) - lon_vec(ii-1)) > pi/2
        discontinuity_idx = [discontinuity_idx ii];
    end
end

for ii = fliplr(discontinuity_idx);
    lon_vec = [lon_vec(1:ii-1); NaN; lon_vec(ii:end)];
    lat_vec = [lat_vec(1:ii-1); NaN; lat_vec(ii:end)];
    geocen_lat_vec = [geocen_lat_vec(1:ii-1); NaN; geocen_lat_vec(ii:end)];
end

figure('OuterPosition', [0 50 hw_pub.figWidth hw_pub.figHeight])
hold on
plot(Map(:,1), Map(:,2))
plot(lon_vec*180/pi, lat_vec*180/pi, 'r')
% plot(lon_vec*180/pi, geocen_lat_vec*180/pi, 'g--')
title('ISS groundtrack for 3 hours')
```

```
xlim([-180,180])
xlabel('Longitude (deg)')
ylim([-90, 90])
ylabel('Latitude (deg)')
axis equal
```

```
Epoch is 10-10-2014 20:51:47.0059 GMT
       = 10-10-2014 14:51:47.0059 MDT
 (used www.epochconverter.com to find day from day-of-year
Must propagate 4.1369 hours (0.17237 days) to reach
 10-10-2014 19:00:00 MDT
```



Published with MATLAB® R2013b