Table of Contents

Problem 1

```
%Morgan Yost
%AERO 557 HW1
clear all
close all
```

Set Up

```
%addpath ../Vallado
global NUM_OBSV
NUM_OBSV = 3;
fid = fopen('input.txt', 'r');
count = 0;
deg_sec = 306/86400;
Re = 6378;
while(count <7)
    count = count +1;
    line = fgetl(fid);
    type = strsplit(line, ' ');
    switch type{1}
        case 'LAT:'
            if(length(type) ~= 2)
                disp('Not enough inputs for Latitude')
                return;
            end
            siteInfo.latSite = str2num(type{2});
            continue;
        case 'LONG:'
            if(length(type) ~= 3)
                length(type)
                disp('Not enough inputs for Longitude')
                return;
            end
            siteInfo.longSite = str2num(type{2});
            siteInfo.longmin = str2num(type{3});
            continue;
        case 'ALT:'
            if(length(type) ~= 2)
                disp('Not enough inputs for Altitude')
```

```
            return;
        end
        siteInfo.H = str2num(type{2});
        continue;
    case 'YEAR:'
        if(length(type) ~= 2)
            disp('Not enough inputs for Year')
            return;
        end
        siteInfo.year(1:3) = str2num(type{2});
        continue;
    case 'MONTH:'
        if(length(type) ~= 2)
            disp('Not enough inputs for Month')
            return;
        end
        siteInfo.month(1:3) = str2num(type{2});
        continue;
    case 'DAY:'
        if(length(type) ~= 2)
            disp('Not enough inputs for Day')
            return;
        end
        siteInfo.day(1:3) = str2num(type{2});
        continue;
    case 'TIME'
        if type{2} == 'RA'
            if(length(type) ~=3)
                disp('Not enough inputs for Day')
                return;
            end
            for i = 1:length(type)
                line = fgetl(fid);
                word = strsplit(line, ' ');
                time = strsplit(word{1}, ':');
                siteInfo.hour(i) = str2num(time{1});
                siteInfo.min(i) = str2num(time{2});
                siteInfo.sec(i) = str2num(time{3});
                alpha(i) = str2num(word{2});
                delta(i) = str2num(word{3});
            end
        elseif type{2} == 'DEC'
            if(length(type) ~=3)
                disp('Not enough inputs for Day')
                return;
            end
            for i = 1:length(type)
                line = fgetl(fid);
                word = strsplit(line, ' ');
                time = strsplit(word, ':');
                siteInfo.hour(i) = str2num(time{1});
                siteInfo.min(i) = str2num(time{2});
                siteInfo.sec(i) = str2num(time{3});
                alpha(i) = str2num(word{3});
```

```
                    delta(i) = str2num(word{2});
                end
            else
                disp('bad data table')
                return;
            end
        otherwise
            continue;
    end
end
fclose(fid);
if count ~= 7
    disp('Missing information for execution. Exiting.')
    return;
end
```

Gauss Angles Only

```
%Site Info Struct
%       H             Altitude of the observation location
%       longSite      Latitude of the observation location
%       latSite       Longitude of the observation location
%       latmin        Minute unit of the latitude of the observation
%                     location
%       year          Year of the observation
%       month         Month of the year of the ovservation
%       day           Day of the month of the observation
%       hour          Hour of the day of the observation
%       min           Minute of the hour of the observation
%       sec           Second of the minute of the observation
[r, v, RSite, qHat, q, tau1, tau3] = gaussOD(alpha, delta, siteInfo);
% Get COEs
[ ~, a, ecc, inc, Omega, ~, ~ ] = RV2COEd( r(:, 2), v);
fprintf('The results of the gauss angles only method are as follows:
\n');
fprintf('r: %f %f %f, \t norm(r): %f \n', r(:, 2), norm(r(:, 2)));
fprintf('v: %f %f %f, \t norm(r): %f \n', v, norm(v));
fprintf('COES: \n a: %f km\n ecc: %f\n inc: %f deg\n RAAN: %f deg\n',
 a,...
    norm(ecc), inc, Omega);
```

```
The results of the gauss angles only method are as follows:
r: 5813.089264 5562.490062 6496.085361,   norm(r): 10340.813677
v: -4.237904 4.393942 1.405317,   norm(r): 6.264301
COES:
 a: 10530.735020 km
 ecc: 0.139085
 inc: 40.045571 deg
 RAAN: 329.861140 deg
```

Extended Gauss

```
[ r2, v2 ] = extendedGaussUV(  RSite, v, qHat, q, tau1, tau3);
```

```
% Get Better COEs
[ ~, a, ecc, inc, Omega, ~, ~ ] = RV2COEd( r2(:, 2), v2);
fprintf('The results of the exended gauss method are as follows:\n');
fprintf('r: %f %f %f, \t norm(r): %f \n', r2(:,2), norm(r2(:,2)));
fprintf('v: %f %f %f, \t norm(r): %f \n', v2, norm(v2));
fprintf('COES: \n a: %f km\n ecc: %f\n inc: %f deg\n RAAN: %f deg\n',
 a,...
    norm(ecc), inc, Omega);
```

The results of the exended gauss method are as follows:
r: 5813.089264 5562.490062 6496.085361,   norm(r): 10340.813677
v: -4.508948 4.780784 1.572110,   norm(r): 6.757073
COES:
a: 12680.318206 km
ecc: 0.237166
inc: 40.045571 deg
RAAN: 329.861140 deg

Double R

```
qHat = qHat*diag(q);
[r2vec, v2vec] = doubleR(qHat(:,1),qHat(:, 2),qHat(:,3), RSite(:,
 1),...
                          RSite(:, 2) ,RSite(:, 3), tau1, tau3);
% Get More COEs
[ p, a, ecc, inc, Omega, w, theta ] = RV2COEd( r2vec, v2vec);
fprintf('The results of the Double r method are as follows:\n');
fprintf('r: %f %f %f, \t norm(r): %f \n', r2vec, norm(r2vec));
fprintf('v: %f %f %f, \t norm(r): %f \n', v2vec, norm(v2vec));
fprintf('COES: \n a: %f km\n ecc: %f\n inc: %f deg\n RAAN: %f deg\n',
 a,...
    norm(ecc), inc, Omega);
```

The results of the Double r method are as follows:
r: 5967.762704 5784.152800 6696.626439,   norm(r): 10673.116742
v: -4.448319 4.722157 1.563498,   norm(r): 6.673143
COES:
a: 13215.526390 km
ecc: 0.246946
inc: 39.957296 deg
RAAN: 330.007181 deg

Explanation

```
fprintf('The results shown above vary due to differences in the
 assumptions \n')
fprintf('that were made to arrive at them. The gauss angles only
 method \n')
fprintf('assumes the angles from the observation are coplanar and does
 not \n')
fprintf('include an iterative scheme to refine the answer. The Double
 R \n')
```

```
fprintf('and Gauss extension, however, do use an iterative scheme and
 it \n')
fprintf('can be seen that this iterations resulted in a much higher
 guess \n')
fprintf('for the semi-major axis. It makes sense that the ecc also
 varies \n')
fprintf('especially between angles only and the other two methods for
 this \n')
fprintf('reason. The inc and RAAN are very similar between the 3
 methods, as \n')
fprintf('as are the r vectors. This just proves that our assumption of
 \n')
fprintf('coplanar observations was correct because these parameters
 are \n')
fprintf('are closely related to the observed right ascention and
 declination. \n')
```

The results shown above vary due to differences in the assumptions that were made to arrive at them. The gauss angles only method assumes the angles from the observation are coplanar and does not include an iterative scheme to refine the answer. The Double R and Gauss extension, however, do use an iterative scheme and it can be seen that this iterations resulted in a much higher guess for the semi-major axis. It makes sense that the ecc also varies especially between angles only and the other two methods for this reason. The inc and RAAN are very similar between the 3 methods, as as are the r vectors. This just proves that our assumption of coplanar observations was correct because these parameters are are closely related to the observed right ascention and declination.

Plots for Sanity Check

figure(1) hold on plot3([0 RSite(1, 1)], [0 RSite(2, 1)], [0 RSite(3, 1)]) plot3([0 RSite(1, 2)], [0 RSite(2, 2)], [0 RSite(3, 2)], 'r') plot3([0 RSite(1, 3)], [0 RSite(2, 3)], [0 RSite(3, 3)], 'g') %% plot3([RSite(1,1) qHat(1, 1)+RSite(1,1)], [RSite(2, 1) qHat(2, 1)+RSite(2, 1)]... , [RSite(3,1) qHat(3, 1)+RSite(3,1)], 'c') plot3([RSite(1, 2) qHat(1, 2)+RSite(1, 2)], [RSite(2, 2) qHat(2, 2)+ RSite(2, 2)],... [RSite(3, 2) qHat(3, 2)+RSite(3, 2)], 'm') plot3([RSite(1, 3) qHat(1, 3)+RSite(1, 3)], [RSite(2, 3) qHat(2, 3)+RSite(2, 3)],... [RSite(3, 3) qHat(3, 3)+RSite(3, 3)], 'y') plot3([0 r(1, 1)], [0 r(2, 1)], [0 r(3, 1)], 'k') plot3([0 r(1, 2)], [0 r(2, 2)], [0 r(3, 2)], 'k') plot3([0 r(1, 3)], [0 r(2, 3)], [0 r(3, 3)], 'k') view(-10, 0)

Published with MATLAB® R2015b

Table of Contents

Problem 2

```
%Morgan Yost
%AERO 557 HW1
clear all
```

Set Up

```
r0 = [15945.34; 0; 0];
r = [12214.83899; 10249.4731; 0];
tm = 1;
dt = 76*60;
mu = 398600;
```

Lambert's Universal

```
[f, g, gdot] = lambertUV(r0, r, dt, tm);
v0_universal = (r-f*r0)/g;
v_universal = (gdot*r-r0)/g;
```

Min Energy

```
[a_min, e_min, t_min_abs, v0_minE] = LambertsMinEnergy( r0, r );
```

Lambert's Gauss

```
[f, g, ~, gdot] = Lambert_Gauss( r0, r, dt, tm );
v0_LM = (r-f*r0)/g;
v_LM = (gdot*r-r0)/g;
```

Izzo Gooding

```
[v0_izzo, v_izzo, extremal_distances, exitflag] = IzzoGooding(...
        r0', r', dt/(60*60*24), 0, mu);
```

Result

```
fprintf('Universal Variable v0: %f %f %f\n', v0_universal');
fprintf('norm: %f\n', norm(v0_universal));
fprintf('Min Energy v0: %f %f %f\n', v0_minE);
fprintf('norm: %f\n', norm(v0_minE));
fprintf('Lambert Gauss v0: %f %f %f\n', v0_LM);
fprintf('norm: %f \n', norm(v0_LM));
fprintf('Izzo Gooding v0: %f %f %f\n', v0_izzo);
fprintf('norm: %f \n', norm(v0_izzo));
```

```
Universal Variable v0: 2.058911 2.915965 0.000000
norm: 3.569589
Min Energy v0: 2.047409 2.924002 0.000000
norm: 3.569548
Lambert Gauss v0: 2.059422 2.915608 0.000000
norm: 3.569593
Izzo Gooding v0: 2.058911 2.915965 0.000000
norm: 3.569589
```

Explanation

```
fprintf('The variance in v vectors from observed r vectors and time
 are very \n')
fprintf('small. The difference comes from minor differences in
 implementation. \n')
fprintf('Lamberts min energy is a direct calculation scheme while the
 other\n')
fprintf('methods are not. It is not suprising to me that the min
 energy and \n')
fprintf('Lambert Gauss methods are so similar because they are both
 geometric \n')
fprintf('methods. \n')
```

```
The variance in v vectors from observed r vectors and time are very
small. The difference comes from minor differences in implementation.
Lamberts min energy is a direct calculation scheme while the other
methods are not. It is not suprising to me that the min energy and
Lambert Gauss methods are so similar because they are both geometric
methods.
```

Published with MATLAB® R2015b