ANDREW GARRETT SAIKO

MAE 593G: GLOBAL POSITIONING SYSTEMS

DR. JASON GROSS

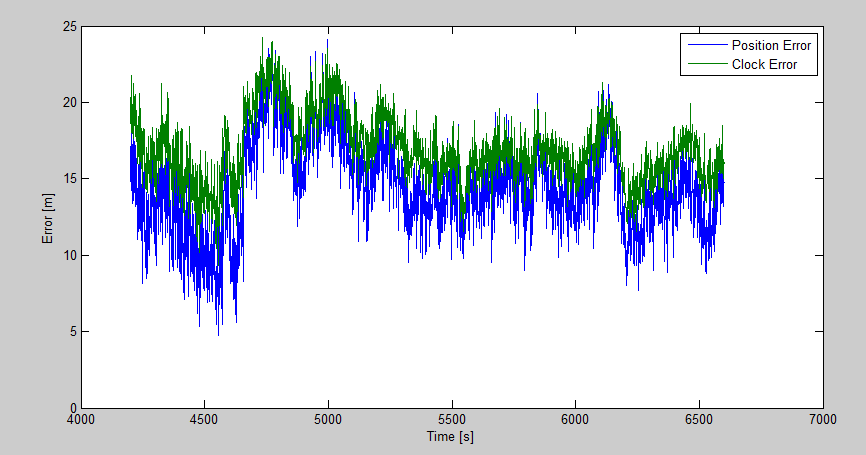
HOMEWORK #1

09/11/2014

# Part 1 – Part 1 – Data Set 1

### XYZ

Beginning with Part 1, a function was developed to estimate the clock and position error for one time epoch. A loop was then configured to store the position and clock error for each time epoch, as well as calculating the final error for each. Figure 1 shows a plot of the final clock error, as well as the final position error for the first data set plotted over each time epoch.

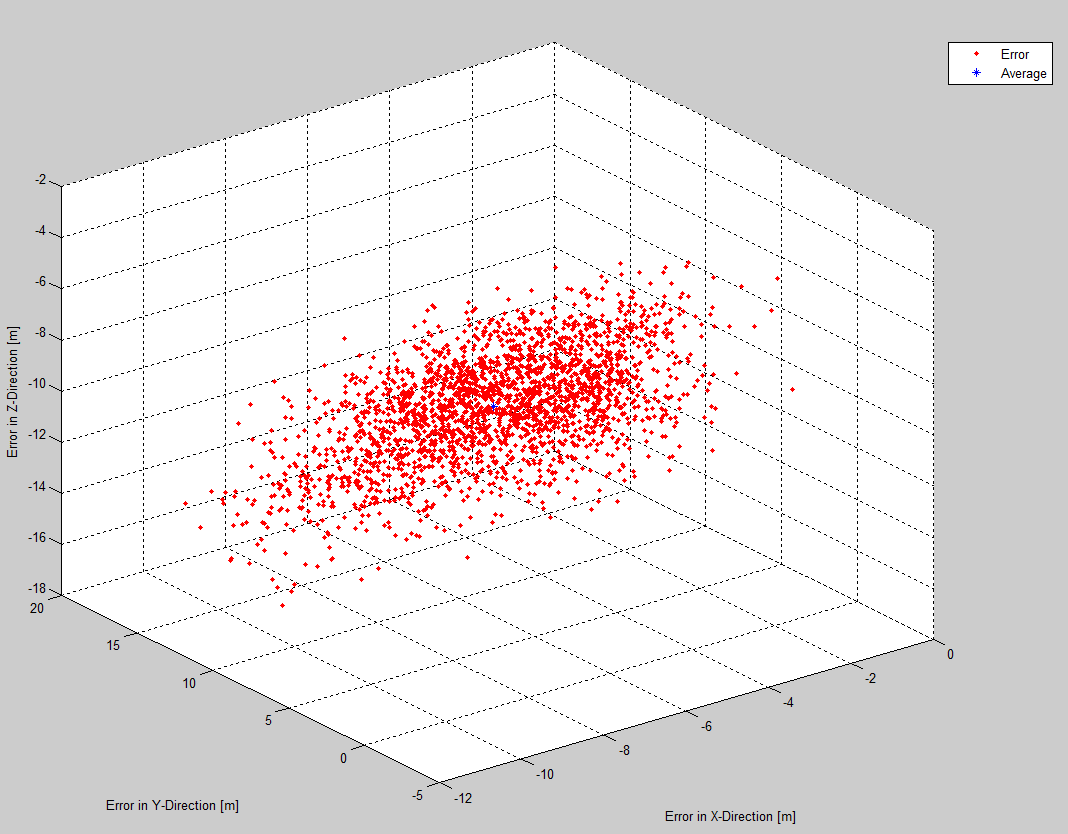


## Figure 1: Position Error and Clock Error vs. Time

Figure 2 shows a scatter plot of the difference in location between the X, Y, and Z coordinates of the actual XYZ location and the estimated XYZ location. The average error for each coordinate is listed in Table 1, and is plotted in Figure 2 in blue.

## Table 1: Average Error for each Coordinate in meters [m]

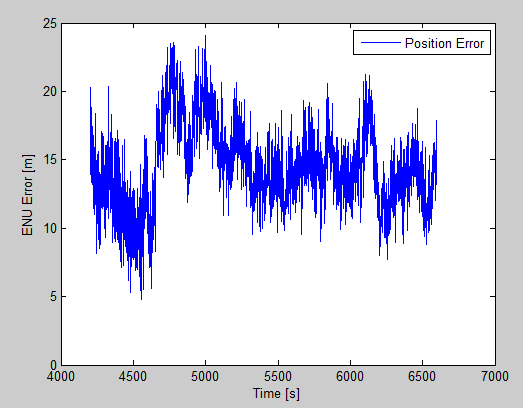
|  |  |  |
| --- | --- | --- |
| X | Y | Z |
| -5.77 | 8.36 | -10.12 |



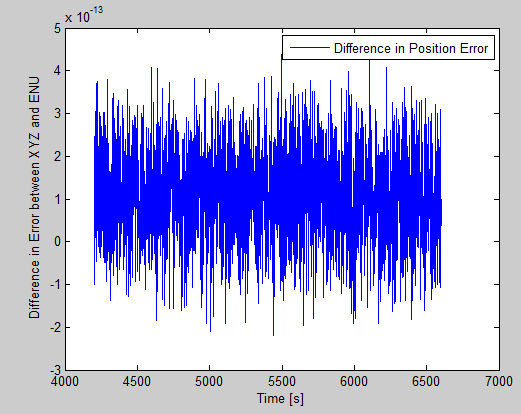
## Figure 2: Difference of Actual XYZ and Estimated XYZ [m]

### ENU

Figure 3 shows the error produced after the coordinate frame had been shifted to ENU. It looks very similar to the ECEF coordinate frame. Figure 4 shows the difference between the errors of the two frames.



## Figure 3: Position Error of ENU Frame vs. Time

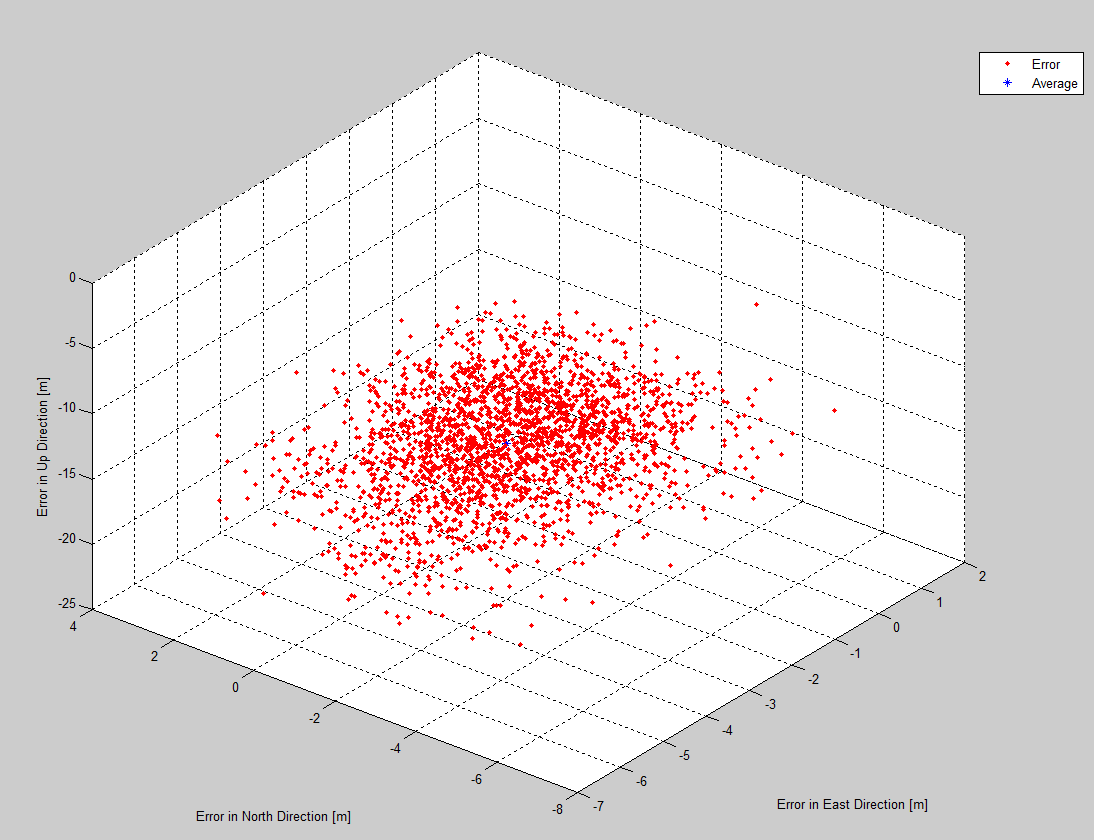


## Figure 4: Difference in XYZ and ENU Position Errors

Looking at the 3D plot for ENU, Table 2 shows the average error for each direction. Figure 5 shows a plot of all of the errors, and a blue tick represents the average.

## Table 2: Average Error for East, North, Up [m]

|  |  |  |
| --- | --- | --- |
| E | N | U |
| -2.73 | -1.71 | -13.97 |

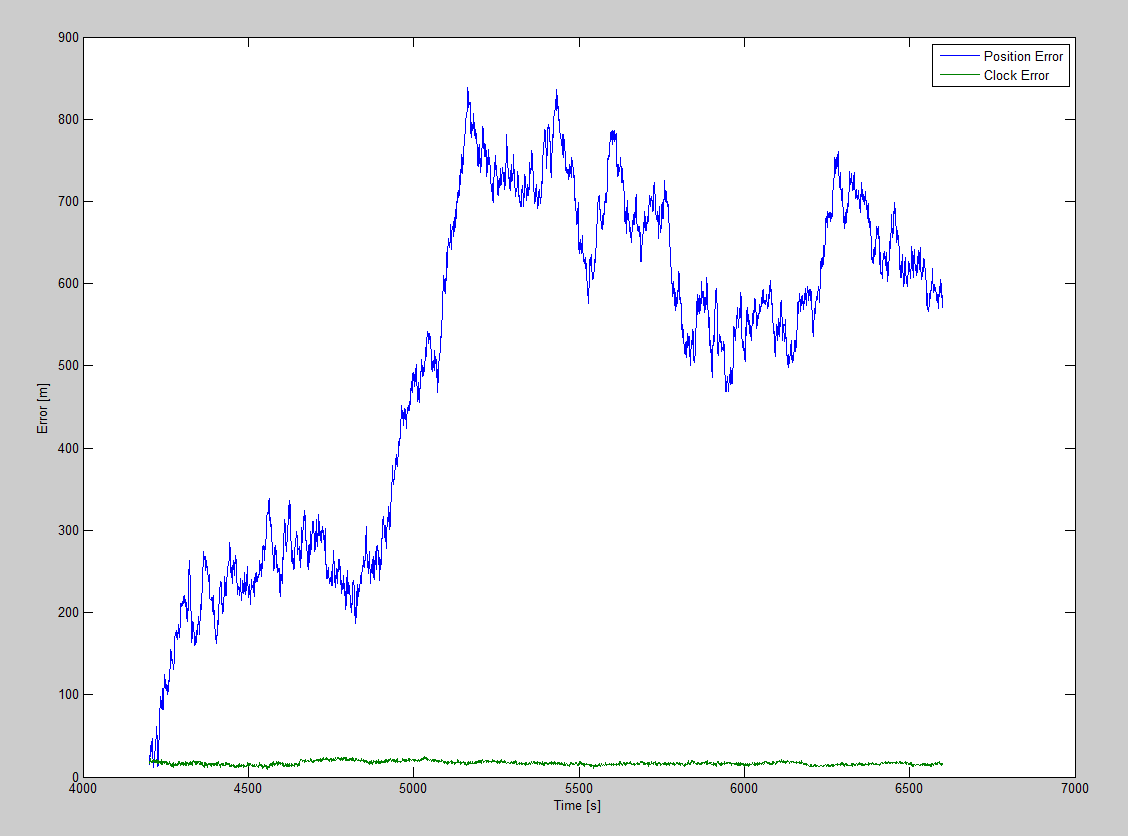


## Figure 5: Difference of Actual ENU and Estimated ENU [m]

# Part 1 – Part 1 – Data Set 2

### XYZ

Looking at the second data set, the position error begins to escalate quickly while the clock error remains relatively low. Figure 6 shows this trend for the position error. This data was produced using the same nominal for each time epoch.

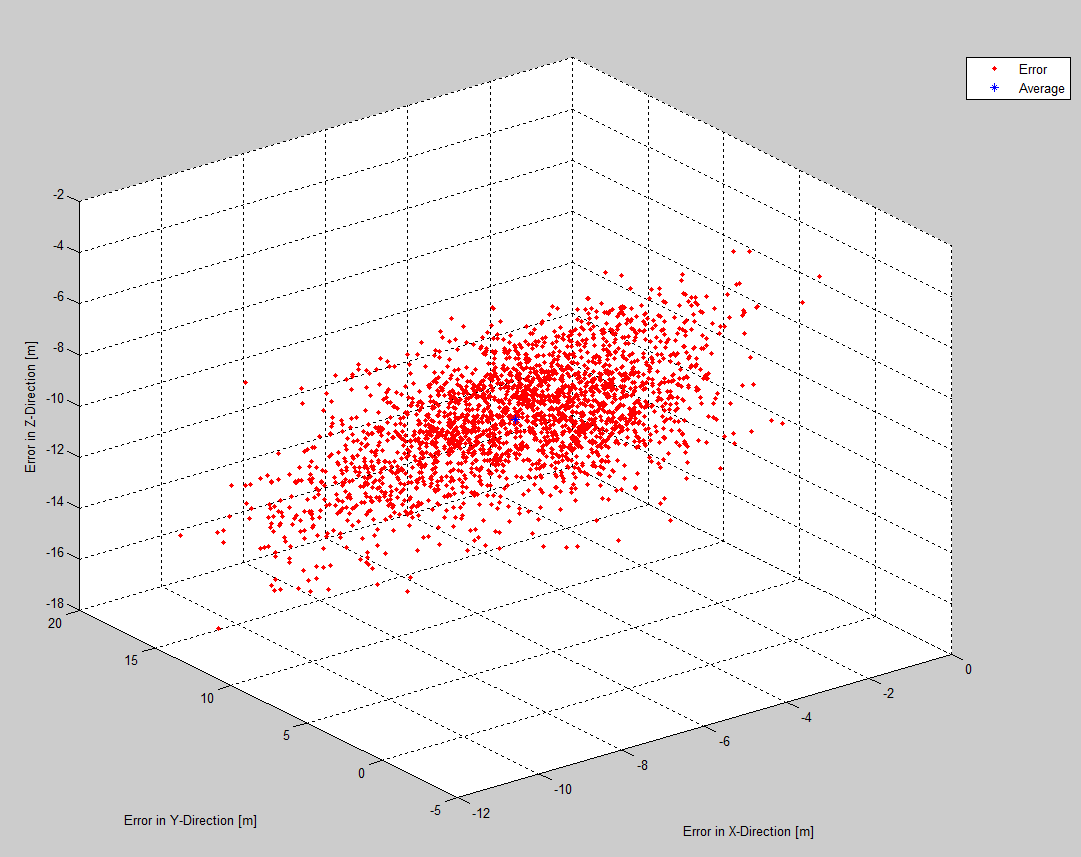


## Figure 6: Position Error and Clock Error vs. Time

Table 3 shows the average error for each coordinate direction after the difference was taken between the actual XYZ locations and the estimated XYZ locations. Figure 7 shows a scatterplot of the errors in the XYZ direction, as well as the mean error between them all.

## Table 3: Average Error for each Coordinate in meters [m]

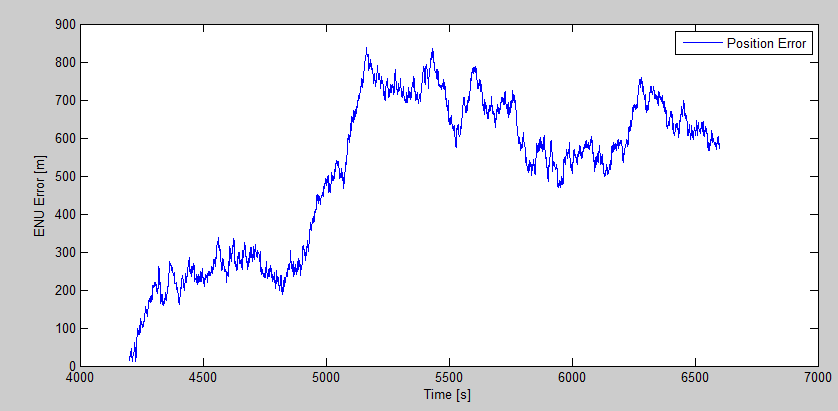
|  |  |  |
| --- | --- | --- |
| X | Y | Z |
| -5.72 | 8.26 | -10.03 |



## Figure 7: Difference of Actual XYZ and Estimated XYZ [m]

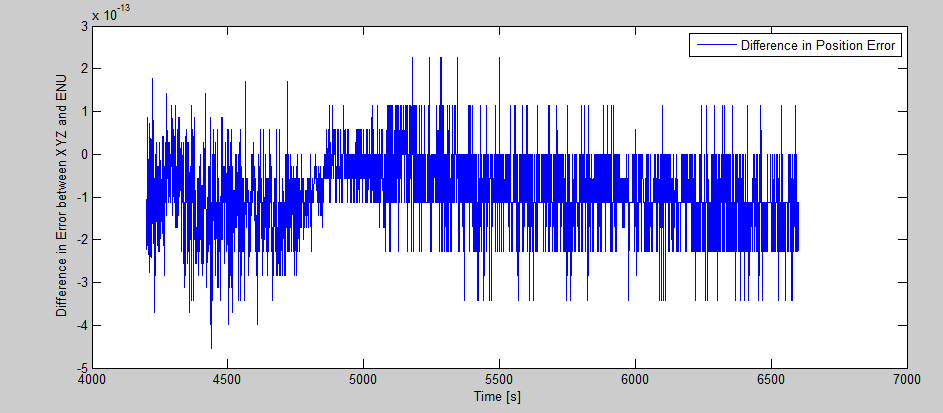
### ENU

Using the provided xyz2enu function, the second data set was able to convert the actual and estimated XYZ to actual and estimated ENU positions. Figure 8 shows the position error in the ENU coordinate frame.



## Figure 8: Position Error in ENU Frame vs. Time

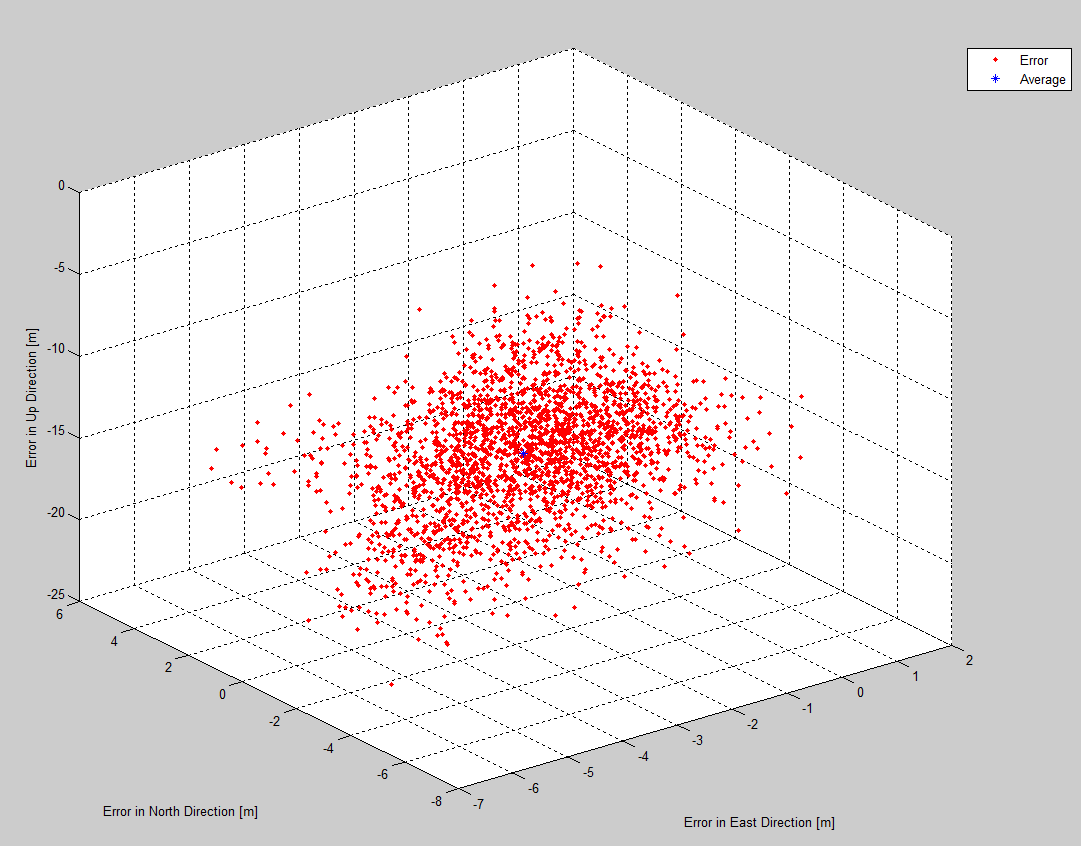
Figure 8 shows the position in the ENU frame vs. time. Figure 9 shows the difference in position error between the ECEF frame and the ENU frame.



## Figure 9: Difference in XYZ and ENU Position Error [m] vs. Time [s]

## Table 4: Average Error for East, North, Up [m]

|  |  |  |
| --- | --- | --- |
| E | N | U |
| -2.72 | -1.71 | -13.82 |

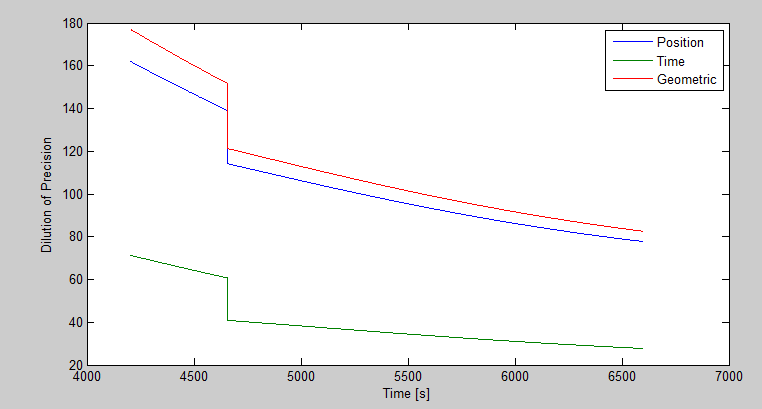


## Figure 10: Difference in Actual ENU and Estimated ENU [m]

Figure 9 shows the difference between each coordinate for the actual and estimated locations. The averages are shown in Table 4, while being plotted in blue on Figure 10

# Part 1 – Part 2 – Data Set 1

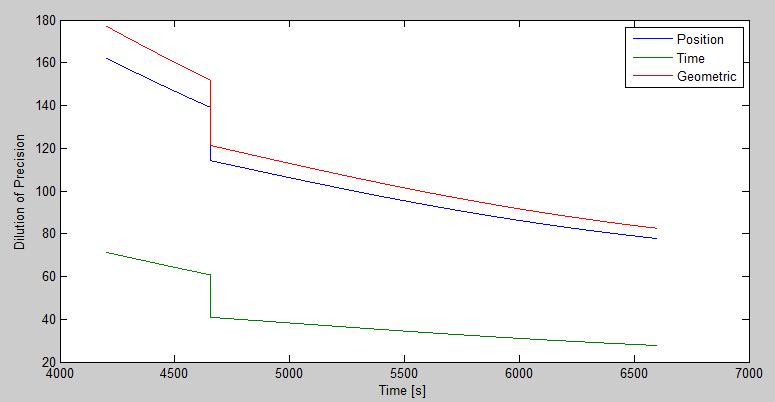
For part 2, the goal was to update the user function to include PDOP, GDOP, and TDOP as outputs. The errors of the position time estimates could then be estimated using the URE sigma provided in the data set. Looking at Figure 11, the graph shows the PDOP, TDOP, and GDOP for the first data set.



## Figure 11: Position, Time, and Geometric Dilution of Precision vs. Time

# Part 1 – Part 2 – Data Set 2

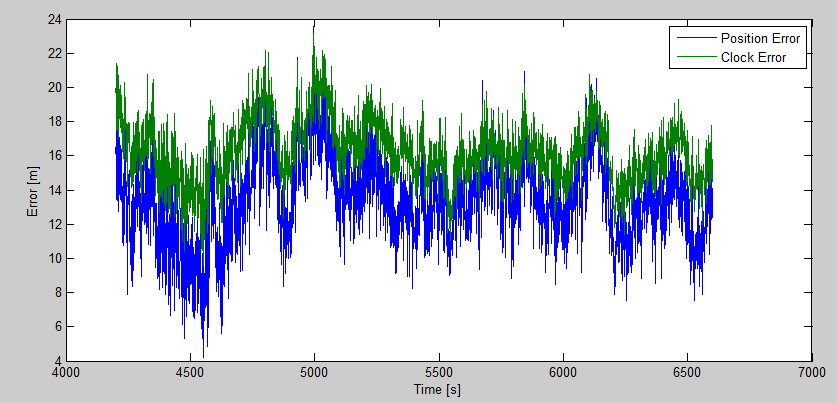
For part 2, the goal was to update the user function to include PDOP, GDOP, and TDOP as outputs. The errors of the position time estimates could then be estimated using the URE sigma provided in the data set. Looking at Figure 12, the graph shows the PDOP, TDOP, and GDOP for the second data set. As you can see, the Dilution of Precision is almost identical for both data sets.



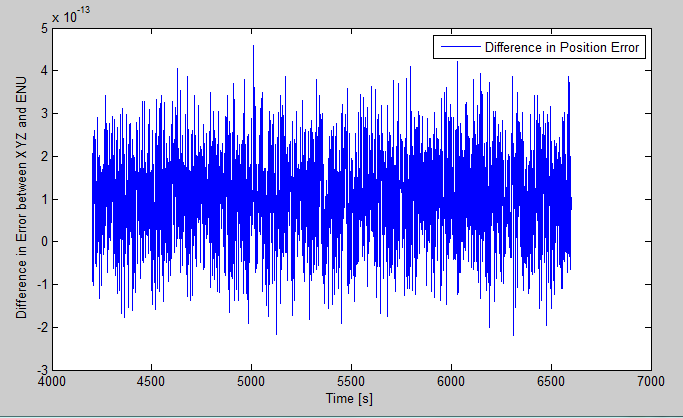
**Figure 12: Position, Time, and Geometric Dilution of Precision vs. Time**

# Part 1 – Part 3 – Data Set 1

After weighting the function and calculating the elevation angles to properly weight each calculated pseudorange, Figure 13 shows the results of the position and clock error. Figure 14 shows the difference between the position errors. Comparing it to the unweighted data, it is similar but not exact.



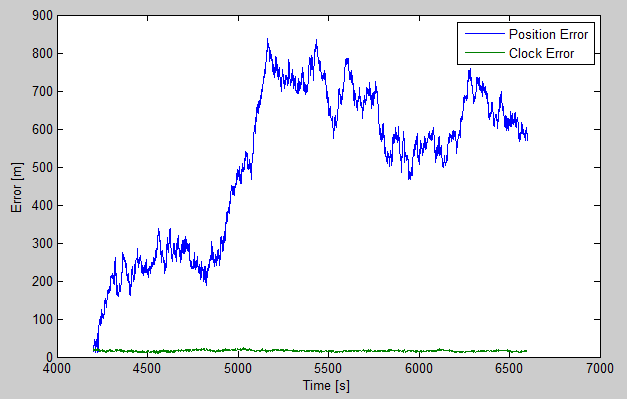
## Figure 13: Weighted Position and Clock Error vs. Time



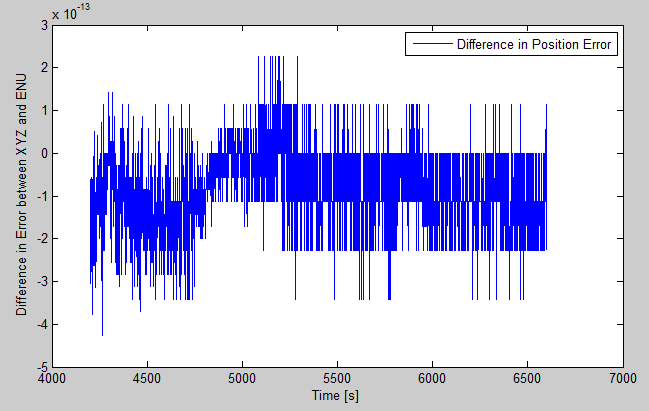
## Figure 14: Difference in Position Error vs. Time

# Part 1 – Part 3 – Data Set 2

Similarly doing the same for data set 2, Figure 15 shows the weighted clock and position error vs. time. Figure 16 shows the difference between errors. Once again the difference is similar to the unweighted errors, but not exactly identical.



## Figure 15: Weighted Position and Clock Error vs. Time



## Figure 16: Difference in Position Error vs. Time

# APPENDICES

% Script File for Function - GPS(nomXYZ,nomCLOCK,pr\_OBS,satsXYZ,nSat)

%% Andrew Saiko - MAE 593 - GPS - Homework #1

format long g

clear

clc

load('i:\MAE 593\HW1\dataSet2.mat')

% This loads the data provided for the homework

%% Creates output variable matrix for each iteration to populate

i=1;

c=1;

x=1;

nom\_XYZ=ones(14,3);

nom\_XYZ(:,1)=nomXYZ(1);

nom\_XYZ(:,2)=nomXYZ(2);

nom\_XYZ(:,3)=nomXYZ(3);

ClockBias\_Est = zeros(2400,1);

XYZ\_Est = zeros(2400,3);

nomCLOCK=clockBiasNom;

I\_Error\_XYZ = zeros(2400,1);

F\_Error\_XYZ = zeros(2400,1);

I\_Error\_CLOCK = zeros(2400,1);

F\_Error\_CLOCK = zeros(2400,1);

enu\_Truth = zeros(14,3,2400);

enu\_Est = zeros(3,2400);

enu\_TRUTH = zeros(3,2400);

PDOP = zeros(2400,1);

TDOP = zeros(2400,1);

GDOP = zeros(2400,1);

sinEL= zeros(2400,1);

tanAZ= zeros(2400,1);

WW=zeros(nSat(i));

W\_Store6=zeros(6,6,456);

W\_Store7=zeros(7,7,1944);

while i<2401;

%% Transforming XYZ\_Truth to ENU\_Truth to establish weighted matrix

x=1;

while x<8

[enu, R] = xyz2enu(satsXYZ(x,:,i),nomXYZ);

enu\_Truth(x,:,i) = enu;

sinEL(x,1) = enu\_Truth(x,3,i)/sqrt(enu\_Truth(x,1,i)^2+enu\_Truth(x,2,i)^2+enu\_Truth(x,3,i)^2);

tanAZ = enu\_Truth(x,1,i)/enu\_Truth(x,2,i);

WW(x,x) = sinEL(x,1);

x=x+1;

end

W=(WW(1:nSat(i),1:nSat(i)));

% if nSat(i)==6;

% W\_Store6(:,:,i)=W;

% elseif nSat(i)==7;

% W\_Store7(:,:,c)=W;

% c=c+1;

% end

%% Truth Inputs for each iteration, i

ClockBias\_Truth = truthClockBias(i);

XYZ\_Truth = truthXYZ(:,i);

%% Giving Inputs and Calculating XYZ/CLOCK Estimates

numSat=nSat(i);

pr\_OBS = prData(1:nSat(i),i);

sats\_XYZ = satsXYZ(1:nSat(i),1:3,i);

c = 299792458; % m/s

% [XYZ\_Estimate] = GPS\_XYZ(nomXYZ,nomCLOCK,pr\_OBS,sats\_XYZ,numSat);

% [Clock\_Bias\_Estimate] = GPS\_CLOCK(nomXYZ,nomCLOCK,pr\_OBS,sats\_XYZ,numSat);

[Clock\_Bias\_Estimate,XYZ\_Estimate,pdop,tdop,gdop] = GPS\_Updated(nomXYZ,nomCLOCK,pr\_OBS,sats\_XYZ,numSat,sigma\_URE,W);

%% Storing Function Estimates in a matrix

ClockBias\_Est(i,1) = Clock\_Bias\_Estimate;

XYZ\_Est(i,:) = XYZ\_Estimate;

PDOP(i,1) = pdop;

TDOP(i,1) = tdop;

GDOP(i,1) = gdop;

%% Calculating Errors and storing in a matrix

I\_Error\_XYZ(i,1) = norm(nomXYZ'-truthXYZ(:,i));

F\_Error\_XYZ(i,1) = norm(XYZ\_Est(i,:)'-truthXYZ(:,1));

I\_Error\_CLOCK(i,1) = norm(nomCLOCK-truthClockBias(1,i));

F\_Error\_CLOCK(i,1) = norm(ClockBias\_Est(i,1)-truthClockBias(1,i));

%% Transforming XYZ -> ENU

[enu, R] = xyz2enu(XYZ\_Estimate,nomXYZ);

enu\_Est(:,i) = enu;

[enu, R] = xyz2enu(XYZ\_Truth,nomXYZ);

enu\_TRUTH(:,i) = enu;

%% Calculated ENU Errors

I\_Error\_XYZ\_ENU(i,1) = norm(nomXYZ'-enu\_TRUTH(:,i));

F\_Error\_XYZ\_ENU(i,1) = norm(enu\_Est(:,i)-enu\_TRUTH(:,1));

%% Iteration Counter

i=i+1;

%% Iterative Method

% if i>1

% nomXYZ = XYZ\_Estimate;

% nomCLOCK = Clock\_Bias\_Estimate;

% end

end

%% Plot PDOP, TDOP, and GDOP

% figure

% plot (time, PDOP, time, TDOP, time, GDOP)

% xlabel('Time [s]');

% ylabel('Dilution of Precision');

% legend('Position','Time','Geometric')

%% Plot the error results - XYZ

figure

plot(time,F\_Error\_XYZ,time,F\_Error\_CLOCK)

legend('Position Error','Clock Error')

xlabel('Time [s]')

ylabel('Error [m]')

%% Plot the error results - ENU

% figure

% plot(time,F\_Error\_XYZ\_ENU)

% legend('Position Error')

% xlabel('Time [s]')

% ylabel('ENU Error [m]')

%% Calculating Difference between XYZ and ENU

figure

Diff = F\_Error\_XYZ - F\_Error\_XYZ\_ENU;

plot(time,Diff)

legend('Difference in Position Error')

xlabel('Time [s]')

ylabel('Difference in Error between XYZ and ENU')

%% Calculate Difference Between Estimate and Truth

% Difference=truthXYZ'-XYZ\_Est;

% Xavg=mean(Difference(:,1));

% Yavg=mean(Difference(:,2));

% Zavg=mean(Difference(:,3));

% AVG\_Error=[Xavg,Yavg,Zavg]

% grid on

% scatter3(Difference(:,1),Difference(:,2),Difference(:,3),'r.')

% xlabel=('Error in X-Direction [m]');

% ylabel=('Error in Y-Direction [m]');

% zlabel=('Error in Z-Direction [m]');

% hold on

% scatter3(Xavg,Yavg,Zavg,'b\*')

% legend('Error','Average')

% hold off

%% Calculating Difference Between Estimate and Truth ENU

% Diff\_ENU = enu\_Truth - enu\_Est;

% Diff\_ENU = Diff\_ENU';

% scatter3(Diff\_ENU(:,1),Diff\_ENU(:,2),Diff\_ENU(:,3),'r.')

% Xavg=mean(Diff\_ENU(:,1));

% Yavg=mean(Diff\_ENU(:,2));

% Zavg=mean(Diff\_ENU(:,3));

% AVG\_Error=[Xavg,Yavg,Zavg]

% grid on

% xlabel=('Error in X-Direction [m]');

% ylabel=('Error in Y-Direction [m]');

% zlabel=('Error in Z-Direction [m]');

% hold on

% scatter3(Xavg,Yavg,Zavg,'b\*')

% legend('Error','Average')

function [Clock\_Bias\_Estimate,XYZ\_Estimate,pdop,tdop,gdop] = GPS\_Updated(nomXYZ,nomCLOCK,pr\_OBS,sats\_XYZ,numSat,sigma\_URE,W)

% This function will take the nominal XYZ location, the nominal clock, the

% observed pseudorange, and the satellite information for each satellite

% and will output the estimate position and estimate clock bias.

% The inputs for this function are listed below with a description.

% nomXYZ - rough guess of GPS location [m]

% nomCLOCK - rough guess of clock bias [m]\*\*

% pr\_OBS - observes pseudoranges set forth by the receiver

% satsXYZ - satellite coordinate information from broadcast navigation message

% nSat - number of satellites used

format long g

c=299792458; % speed of light [m/s]

% pre allocate sizes

prComputed=zeros(numSat,1);

uNom2Sat=zeros(numSat,3);

for i=1:numSat;

% calculate computed or expected psuedorange

prComputed(i)=norm(sats\_XYZ(i,:)-nomXYZ)+nomCLOCK\*c;

% calculate the unit vectors from nominal position

% to satellite location

% which make up the partials

% of the observation model

uNom2Sat(i,:)=(sats\_XYZ(i,:)-nomXYZ)/norm(sats\_XYZ(i,:)-nomXYZ);

end

% Add the weighted matrix

% W = diag(numSat);

% add the partial of the clock bias, observation model

G=horzcat(uNom2Sat,ones(numSat,1));

% form the innovation vector

deltaRho=prComputed-pr\_OBS;

% find the delta solution

dX=inv(G'\*W\*G)\*G'\*W\*deltaRho;

H = inv(G'\*G);

% calculate and solve for covariance matrix

P\_error = sigma\_URE^2 \* eye(numSat);

cov\_dX = sigma\_URE^2 \* H;

sigmaX = cov\_dX(1,1);

sigmaY = cov\_dX(2,2);

sigmaZ = cov\_dX(3,3);

sigmat = cov\_dX(4,4);

% Calculate different dilutions of precision

pdop = sqrt(sigmaX^2+sigmaY^2+sigmaZ^2);

tdop = sqrt(sigmat^2);

gdop = sqrt(sigmaX^2+sigmaY^2+sigmaZ^2+sigmat^2);

% the clock bias has a negative sign that must be handled

XYZ\_Estimate=nomXYZ+dX(1:3)';

Clock\_Bias\_Estimate=nomCLOCK+-1\*dX(4);

end