AAE 575

Introduction to Satellite Navigation and Positioning

Homework 3

November 11, 2011

Josh Wildey

**Objective:**

The first task at hand is to compute the ionosphere-free measurement calculations of the pseudorange and pseudorange rate using dual frequency measurement techniques. It is then required to estimate the frequency and clock biases using the given data. Then, from these measurements and calculations, determine the stability of the oscillator and use the carrier phase to smooth out the pseudorange measurement. The next task is to implement the Klobuchar model to estimate the Ionosphere delay and to calculate the corrected model and compare this model to the pseudorange measurements. Lastly, given another set of data, the tropospheric delay needs to be calculated by implementing the Sasstamoinen and Hopfield Models. These models are then to be used for comparing the mapping function between flat-earth, Misra and Enge, and Chao Mapping function coefficients.

**Given Data:**

*files: rawdata.sv13, orbit.sv13, dualfreq.sv1, DH010600.MET*

*ECEF Location of CORS Station:*

*X = -1288337.0539 m*

*Y = -4721990.4483 m*

*Z = 4078321.6617 m*

|  |  |  |  |
| --- | --- | --- | --- |
| *α0*  *0.028 x 10-6* | *α1*  *-0.007 x 10-6* | *α2*  *-0.119 x 10-6* | *α3*  *0.199 x 10-6* |
| *β0*  *137.0 x 103* | *β1*  *-49.0 x 103* | *β2*  *-131.0 x 103* | *β3*  *-262.0 x 103* |

*ECEF Location of Durmid Hill GPS Station:*

*Φ =33.390° N*

*λ = 115.79° W*

*h = 0 m*

**Problem 1:**

Part 1:

To calculate the ionosphere free pseudoranges the following equations were used:

Solving for A leads to a value of:

Plugging this back into one of the equations and solving to the pseudorange, , yields:

Similarly for the ionosphere free carrier phase scaled to the L1 frequency yields:

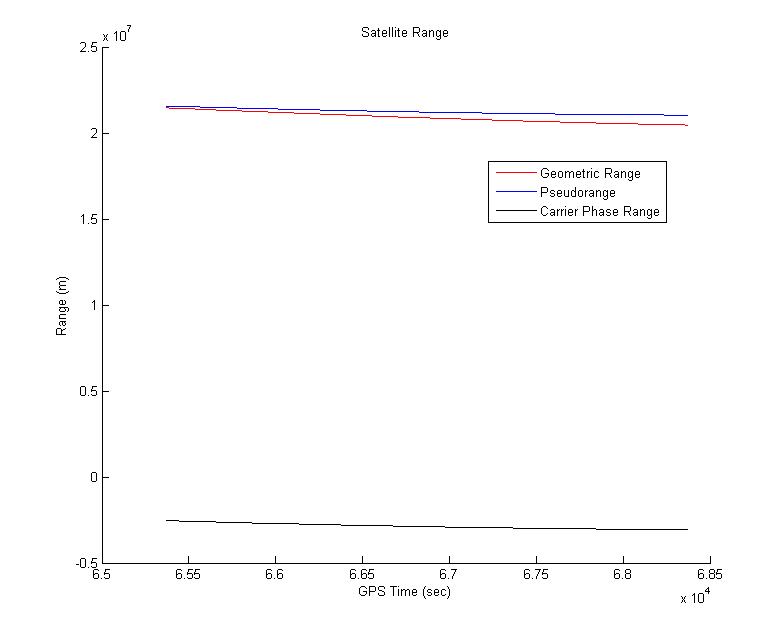


Figure 1‑ Satellite Range

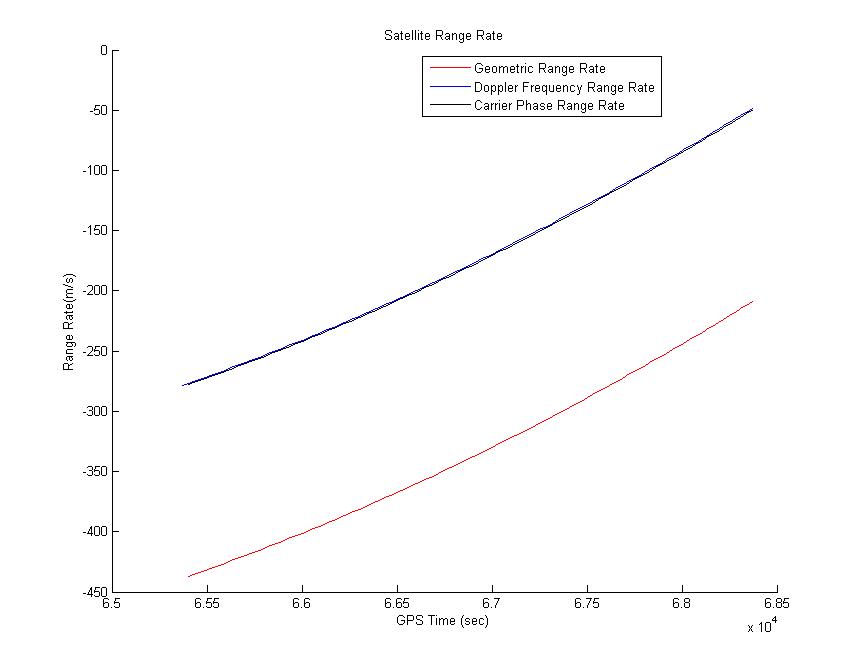


Figure 1- Satellite Range Rate

Part 2:

To determine the oscillator frequency bias, the following equation was used:

To calculate the Line of Sight velocity the Frequency bias was multiplied by the wavelength of the frequency. Results can be seen in figures 1-3 and 1-4.

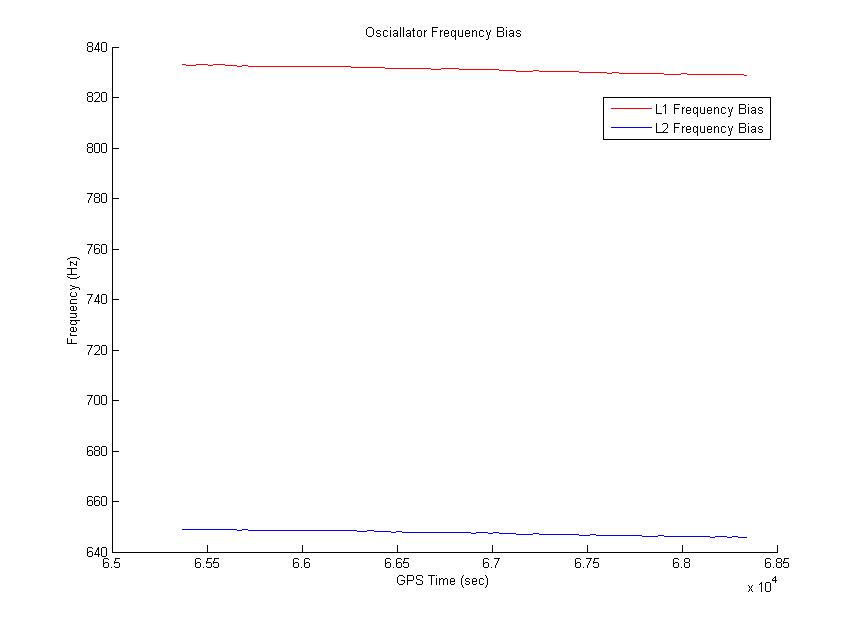


Figure 1‑ Oscillator Frequency Bias

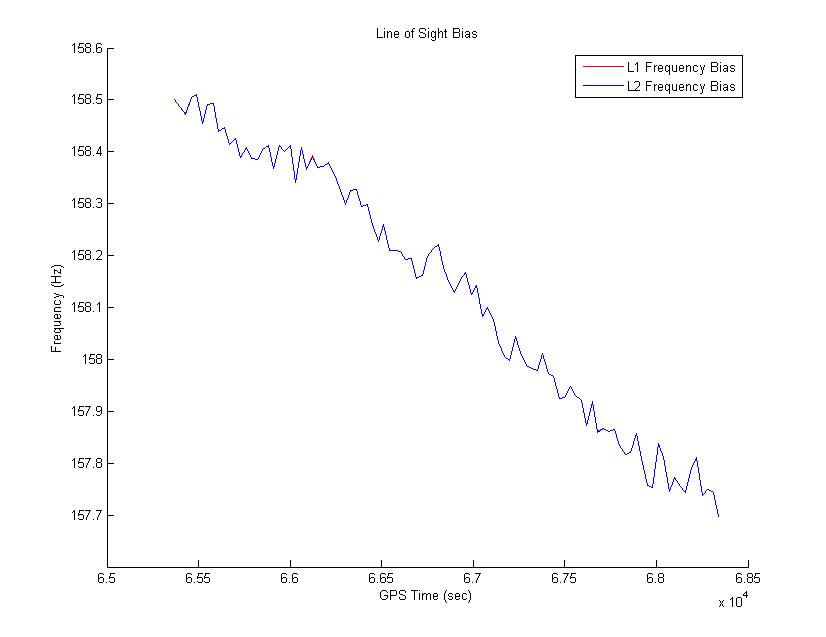
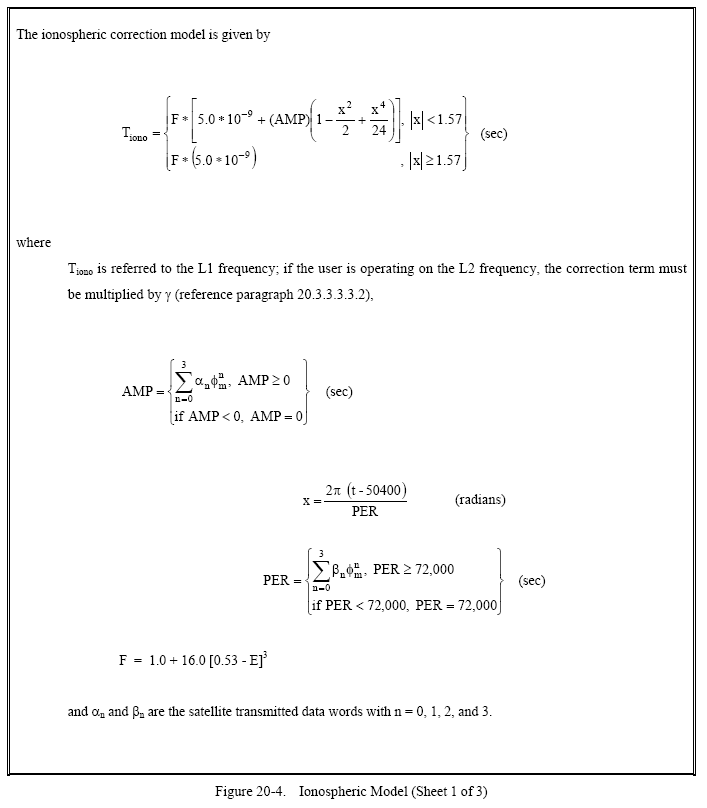
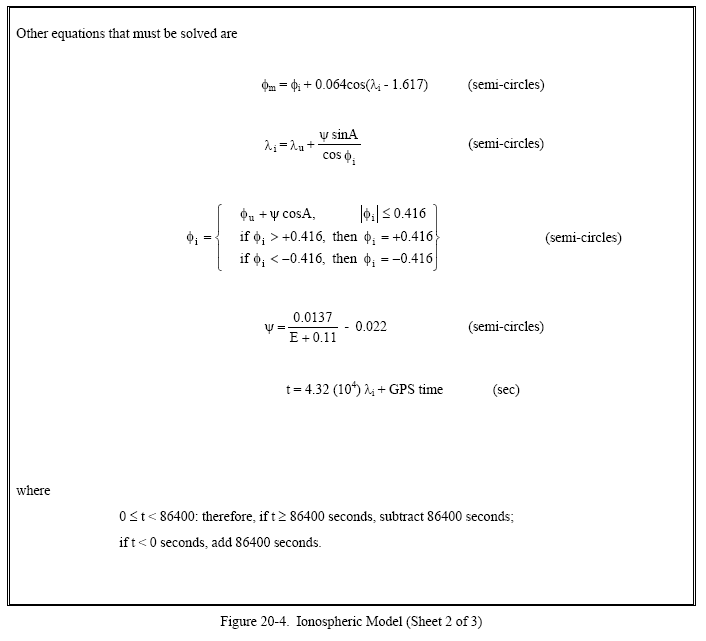


Figure 1‑ Line of Sight Bias

**Problem 2 and 3:**

To implement the Klobuchar model the following equations were used and were referenced from the IS-GPS-200E manual.





The following figure show the results.

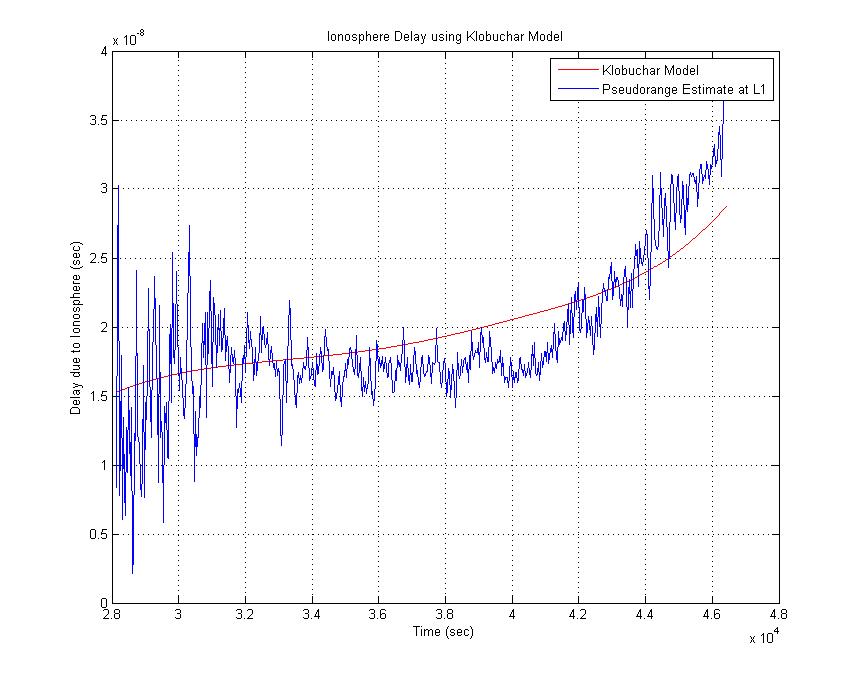


Figure 1‑ ‑ Klobuchar Model and Pseudorange estimation

The pseudorange was estimated using the following equations:

Although the pseudorange estimation is very noisy, the Klobuchar model can still be seen to model the pseudorange with the same general curvature. It is not perfect, but represents a decent model.

**Problem 4:**

To implement the Sasstamoinen model the following equations were used:

To implement the Hopfield Model the following equations were used:

The results are as follows:

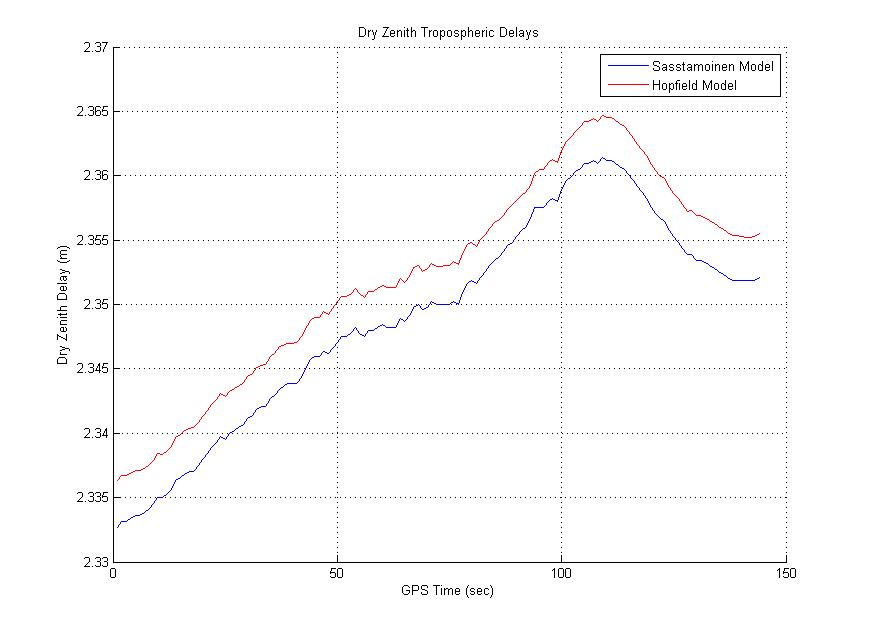


Figure 1‑Error! No text of specified style in document. Dry Zenith Delays

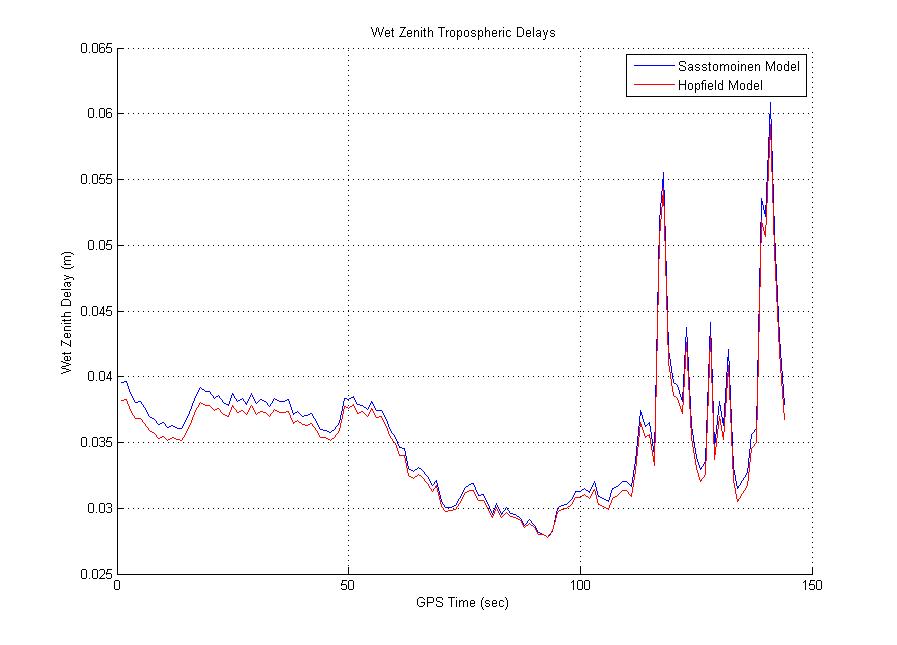


Figure 1‑ Wet Zenith Delays

Both models of tropospheric delays are very similar with only minor offsets in their values and could be used interchangeably for correct results.

**Problem 5:**

To compare the three different mapping functions the following coefficients and equations were used:

For simple flat-earth Mapping:

For Misra and Enge Mapping:

For Chao Mapping:

The total tropospheric delay can be represented by the following equation:

The total tropospheric delay was calculated for elevation angles of 5°, 15°, 45°, and 85° and can be seen in the following figures.

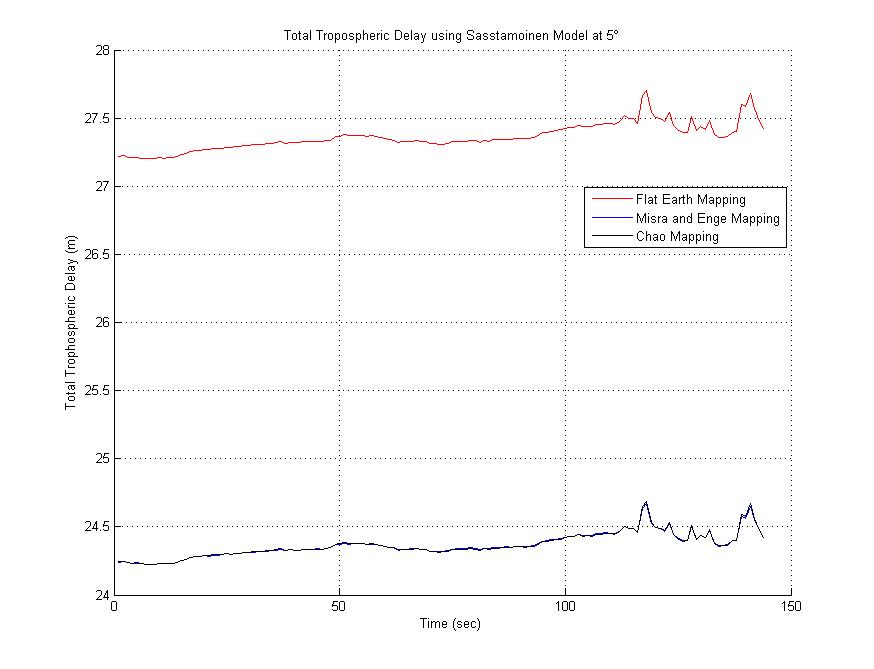


Figure 1‑

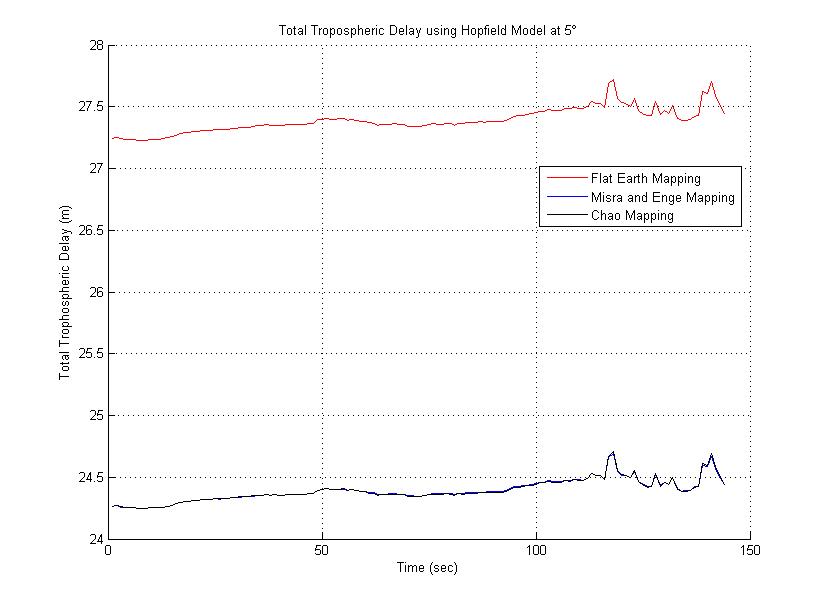
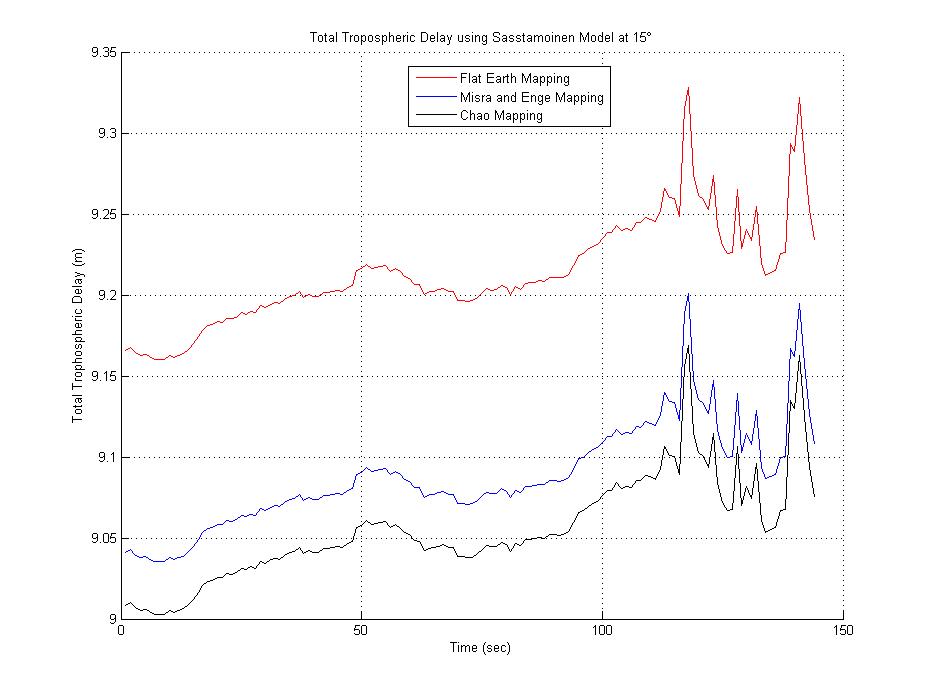


Figure 1‑



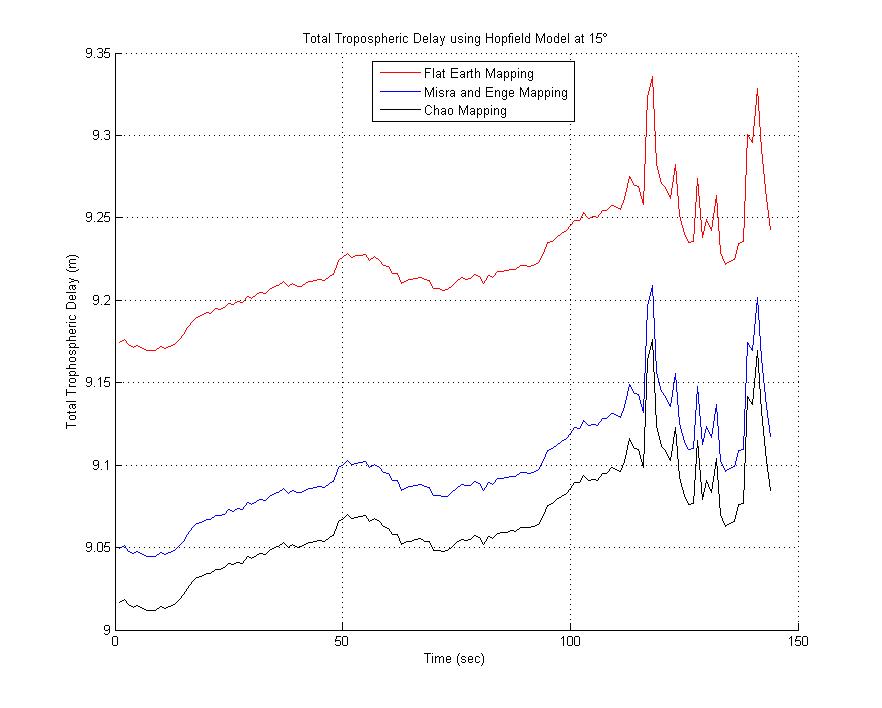


Figure 1‑

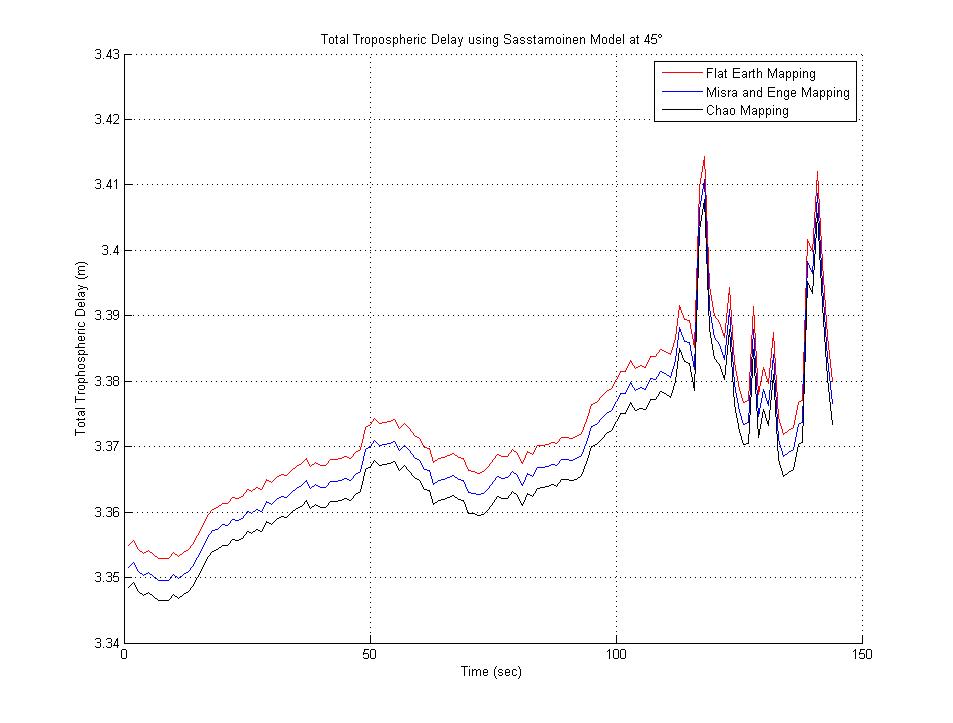


Figure 1‑

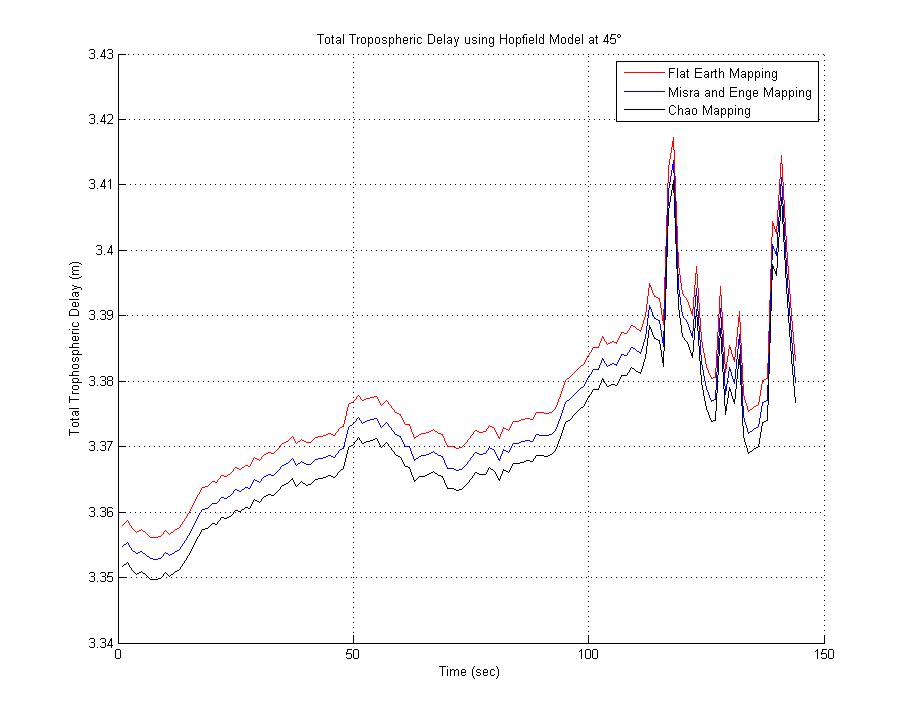


Figure 1‑

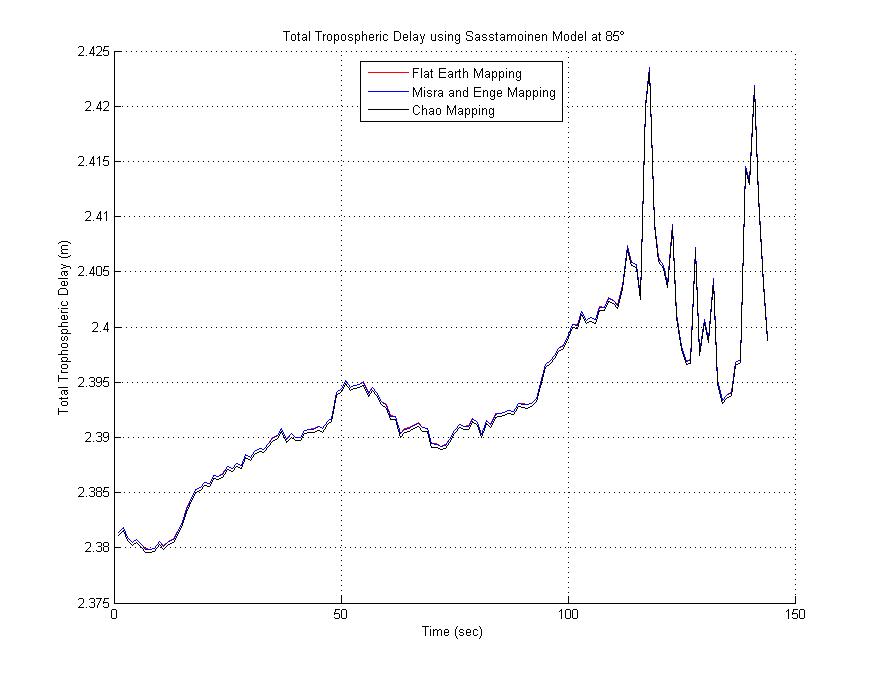


Figure 1‑

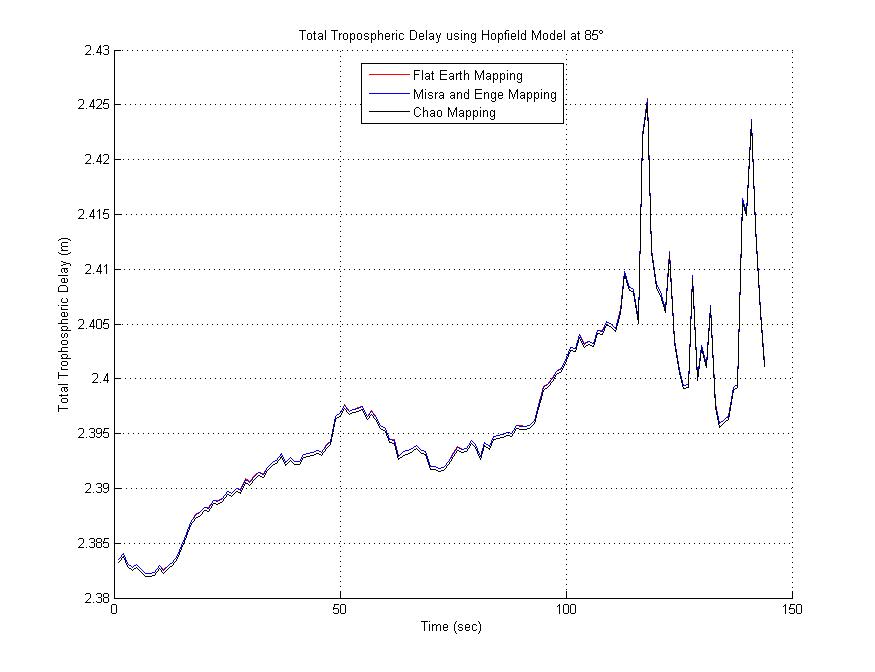


Figure 1‑

As shown in these figures, as the elevation angle gets closer and closer to 90° the different models become almost identical. At low elevation angles there are high error rates, especially with the flat earth mapping function as it is known to produce bad results below 15°. It is also observed that the actual delay decreases significantly as the elevation angle increases, again becoming more accurate and precise as the elevation angle increases. In the range between 5° and 85°, it is seen that the models still hold very similar shape and are just offset from each other.

**Appendix A: Code**

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Author: Josh Wildey

% Class: AAE57500

% Homework 3 - 11/4/2011

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

close all

clear all

clc

%%%%%%%%% Problem 1: %%%%%%%%%

% Load data and define Global Variables

load orbit0.sv13 %Load Data Set

load rawdata.sv13 %Load Data Set

load dualfreq.sv1 %Load Data set

load DH010600.MET %Load Data set

xr = -1288337.0539; %x-position of base station (m)

yr = -4721990.4483; %y-position of base station (m)

zr = 4078321.6617; %z-position of base station (m)

c = 299792458; %Speed of light (m/s)

fL1 = 1575.42e6; %Frequency of L1 (Hz)

fL2 = 1227.60e6; %Frequency of L2 (Hz)

lambdaL1 = c/fL1; %Wavelength of L1 (m)

lambdaL2 = c/fL2; %Wavelength of L2 (m)

GPStime = rawdata(:,1); %SV 13 GPS time from given data set (s)

L1\_Prange = rawdata(:,2); %L1 Pseudorange from given data set (m)

L2\_Prange = rawdata(:,3); %L2 Pseudorange from given data set (m)

L1\_Cphase = rawdata(:,4); %L1 Carrier Phase from given data set (cycles)

L2\_Cphase = rawdata(:,5); %L2 Carrier Phase from given data set (cycles)

L1\_DopFreq = rawdata(:,6); %L1 Doppler Freqeuncy from data set (Hz)

L2\_DopFreq = rawdata(:,7); %L2 Doppler Frequency from data set (Hz)

GPStime\_rx = orbit0(:,1); %GPS time of signal reception (s)

% Calculate True Geometric Range

geo\_range = sqrt((xr-orbit0(:,2)).^2+(yr-orbit0(:,3)).^2+(zr-orbit0(:,4)).^2);

% Calculate True Geometric Range Rate

geo\_range\_rate = diff(geo\_range)./diff(GPStime);

% Calculate Ionosphere Free Pseudorange

ionfree\_Prange = (fL1^2/(fL1^2-fL2^2))\*L1\_Prange - (fL2^2/(fL1^2-fL2^2))\*L2\_Prange;

% Calculate Phi\*

phistar\_L1 = (fL1^2/(fL1^2-fL2^2))\*L1\_Cphase-(fL1\*fL2/(fL1^2-fL2^2))\*L2\_Cphase;

% Calculate Carrier Phase Pseudorange

Cphase\_range\_L1 = lambdaL1\*phistar\_L1;

% Calculate Range Rate from Doppler Frequency

DopFreq\_RangeRate = -lambdaL1\*L1\_DopFreq;

% Calculate Range Rate from Phi\*

phistar\_L1\_RangeRate = lambdaL1\*(diff(phistar\_L1)./diff(GPStime));

figure(1)

hold on

plot(GPStime,geo\_range,'r')

plot(GPStime,ionfree\_Prange,'b')

plot(GPStime,Cphase\_range\_L1,'k')

title('Satellite Range'),xlabel('GPS Time (sec)'),ylabel('Range (m)')

legend('Geometric Range', 'Pseudorange', 'Carrier Phase Range')

hold off

figure(2)

hold on

plot(GPStime(2:length(GPStime)),geo\_range\_rate,'r')

plot(GPStime,DopFreq\_RangeRate,'b')

plot(GPStime(2:length(GPStime)),phistar\_L1\_RangeRate,'k')

title('Satellite Range Rate'),xlabel('GPS Time (sec)'),ylabel('Range Rate(m/s)')

legend('Geometric Range Rate', 'Doppler Frequency Range Rate', 'Carrier Phase Range Rate')

hold off

% Calculate Frequency Bias (Part 2)

fL1\_bias = -L1\_DopFreq(1:length(geo\_range\_rate)) - geo\_range\_rate/lambdaL1;

fL2\_bias = -L2\_DopFreq(1:length(geo\_range\_rate)) - geo\_range\_rate/lambdaL2;

% Calculate Line of Sight Velocity (Part 2)

LoS\_fL1 = lambdaL1\*fL1\_bias;

LoS\_fL2 = lambdaL2\*fL2\_bias;

figure(3)

hold on

plot(GPStime(1:length(fL1\_bias)),fL1\_bias,'r')

plot(GPStime(1:length(fL2\_bias)),fL2\_bias,'b')

title('Osciallator Frequency Bias'),xlabel('GPS Time (sec)'),ylabel('Frequency (Hz)')

legend('L1 Frequency Bias','L2 Frequency Bias')

hold off

figure(4)

hold on

plot(GPStime(1:length(LoS\_fL1)),LoS\_fL1,'r')

plot(GPStime(1:length(LoS\_fL2)),LoS\_fL2,'b')

title('Line of Sight Bias'),xlabel('GPS Time (sec)'),ylabel('Frequency (Hz)')

legend('L1 Frequency Bias','L2 Frequency Bias')

hold off

%%%%%%%%% Problem 2: %%%%%%%%%

El = dualfreq(:,3)/180; %Elevation angle in units of semi-circles

Az = dualfreq(:,2)/180; %Azimuth angle in units semi-circles

alpha = [0.028e-6,-0.007e-6,-0.119e-6,0.119e-6];

beta = [137e3,-49e3,-131e3,-262e3];

% Calculate Obliquity factor (dimensionless)

F = 1.0 + 16.0\*(.53-El).^3;

% Calculate Earth's central angel between user position and earth

% projection of ionospheric intersection point (semi-circles)

psi = .0137./(El+.11) - .022;

% Convert Base station Earth Center Earth Fixed coordinates to latitude,

% longitude, altitude

lla = ecef2lla([xr yr zr]);

latR = lla(1)/180; %Latitude of Rx in SC units

longR = lla(2)/180; %Longitude of Rx in SC units

altR = lla(3); %Altitude of Rx in m

% Calculate Geodetic latitude of the earth projection of the ionospheric

% instersection point with anomoly correction based on IS-GPS-200E Spec

phii = latR + psi.\*cosd(Az);

for q = 1:length(phii)

if phii(q) > .416

phii(q) = .416;

elseif phii(q) < -.416

phii(q) = -.416;

else

phii(q) = phii(q);

end

end

% Calculate Geodetic longitude of earth projection of the ionospheric

% intersection point

lambdai = longR + psi.\*sind(Az)./cosd(phii);

% Calculate Geomagnetic latitude of earth projection of ionospheric

% intersection

phim = phii + .064.\*cosd(lambdai-1.617);

% Calculate local time with autocorrection based on IS-GPS-200E Spec

t = 4.32e4.\*lambdai + dualfreq(:,1);

for q = 1:length(t)

if t(q) >= 86400

t(q) = t(q) - 86400;

elseif t(q) < 0

t(q) = t(q) + 86400;

end

end

% Calculate A4 with autocorrection based on IS-GPS-200E Spec

PER = beta(1).\*(phim.^0) + beta(2).\*(phim.^1) + beta(3).\*(phim.^2) + beta(4).\*(phim.^3);

for q = 1:length(PER)

if PER(q) < 72000

PER(q) = 72000;

end

end

% Calculate phase (radians)

x = (2\*pi\*(t-50400))./PER;

% Calculate A2 with autocorrection based on IS-GPS-200E Spec

AMP = alpha(1).\*(phim.^0) + alpha(2).\*(phim.^1) + alpha(3).\*(phim.^2) + alpha(4).\*(phim.^3);

for q = 1:length(AMP)

if AMP(q) < 0

AMP(q) = 0;

end

end

%Ionospheric correction model based on IS-GPS-200E Spec

Tiono = zeros(size(x));

for q = 1:length(t)

if abs(x(q)) >= 1.57

Tiono(q) = F(q)\*5e-9;

elseif abs(x(q)) < 1.57

Tiono(q) = F(q)\*(5e-9+AMP(q)\*(1 - x(q)^2/2 + x(q)^4/24));

end

end

figure(5)

plot(t,Tiono,'r')

title('Ionosphere Delay using Klobuchar Model'),xlabel('Time (sec)'), ylabel('Delay due to Ionosphere (sec)')

grid on

hold on

%%%%%%%%% Problem 3: %%%%%%%%%

% Compute the Value of A

A = (fL1^2\*fL2^2/(fL2^2-fL1^2)).\*(dualfreq(:,4)-dualfreq(:,5));

% Compute Ionospheric Delay

tau\_iono = A/(c\*fL1^2);

plot(t,tau\_iono)

legend('Klobuchar Model','Pseudorange Estimate at L1')

hold off

%%%%%%%%% Problem 4: %%%%%%%%%

%Local Variable Definitions

year = DH010600(:,1);

month = DH010600(:,2);

day = DH010600(:,3);

hour = DH010600(:,4);

minute = DH010600(:,5);

sec = DH010600(:,6);

pressure = DH010600(:,7); % Surface Atmospheric Pressure (mbar)

temp = DH010600(:,8) + 273.15; % Surface Atmospheric Temperature (deg K)

humidity = DH010600(:,9)/100; % Relative Humidity

phi = 33.39; % Degrees N (Latitude)

lambda = 115.79; % Degrees W (Longitude)

h = 0; % Altitude (meters)

% Calculate Water Vapor Partial Pressure

e0 = 6.108\*humidity.\*exp((17.15\*temp - 4684)./(temp - 38.5));

% Calculate Sasstamoinen Zenith Delays

Sasst\_dry = .0022777\*(1 + .0026\*cosd(2\*phi) + .00028\*h).\*pressure;

Sasst\_wet = .0022777\*(1255./temp + .05).\*e0;

% Calculate Heights for Hopfield Model

hd = 40136 + 148.72.\*(temp - 273.16);

hw = 11000;

% Calculate Hopfield Model

hop\_dry = 77.64e-6.\*pressure.\*hd./(5\*temp);

hop\_wet = .373\*e0\*hw./(5\*temp.^2);

figure(6)

hold on, grid on

plot(Sasst\_dry,'b')

plot(hop\_dry,'r')

title('Dry Zenith Tropospheric Delays'),xlabel('GPS Time (sec)'),ylabel('Dry Zenith Delay (m)')

legend('Sasstamoinen Model','Hopfield Model')

figure (7)

hold on, grid on

plot(Sasst\_wet,'b')

plot(hop\_wet,'r')

title('Wet Zenith Tropospheric Delays'),xlabel('GPS Time (sec)'),ylabel('Wet Zenith Delay (m)')

legend('Sasstomoinen Model','Hopfield Model')

%%%%%%%%% Problem 5: %%%%%%%%%

% Local Variable Definitions

elev = [5 15 45 85];

% Calculate Flat-Earth Mapping

m = 1./sind(elev);

flat\_delay\_model\_sasst = zeros(length(Sasst\_dry),length(elev));

flat\_delay\_model\_hop = zeros(length(Sasst\_dry),length(elev));

for q = 1:length(elev)

flat\_delay\_model\_sasst(:,q) = Sasst\_wet\*m(q) + Sasst\_dry\*m(q);

flat\_delay\_model\_hop(:,q) = hop\_wet\*m(q) + hop\_dry\*m(q);

end

% Calculate Misra and Enge Mapping

mME = 1./(sqrt(1-(cosd(elev)./1.001).^2));

misra\_delay\_model\_sasst = zeros(length(Sasst\_dry),length(elev));

misra\_delay\_model\_hop = zeros(length(Sasst\_dry),length(elev));

for q = 1:length(elev)

misra\_delay\_model\_sasst(:,q) = Sasst\_wet\*mME(q) + Sasst\_dry\*mME(q);

misra\_delay\_model\_hop(:,q) = hop\_wet\*mME(q) + hop\_dry\*mME(q);

end

% Calculate Chao Mapping

mchao\_dry = 1./(sind(elev)+.00143./(tand(elev)+.0445));

mchao\_wet = 1./(sind(elev)+.00035./(tand(elev)+.017));

chao\_delay\_model\_sasst = zeros(length(Sasst\_dry),length(elev));

chao\_delay\_model\_hop = zeros(length(Sasst\_dry),length(elev));

for q = 1:length(elev)

chao\_delay\_model\_sasst(:,q) = Sasst\_wet\*mchao\_wet(q) + Sasst\_dry\*mchao\_dry(q);

chao\_delay\_model\_hop(:,q) = hop\_wet\*mchao\_wet(q) + hop\_dry\*mchao\_dry(q);

end

figure(8)

hold on, grid on

plot(flat\_delay\_model\_sasst(:,1),'r')

plot(misra\_delay\_model\_sasst(:,1),'b')

plot(chao\_delay\_model\_sasst(:,1),'k')

title('Total Tropospheric Delay using Sasstamoinen Model at 5\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(9)

hold on, grid on

plot(flat\_delay\_model\_hop(:,1),'r')

plot(misra\_delay\_model\_hop(:,1),'b')

plot(chao\_delay\_model\_hop(:,1),'k')

title('Total Tropospheric Delay using Hopfield Model at 5\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(10)

hold on, grid on

plot(flat\_delay\_model\_sasst(:,2),'r')

plot(misra\_delay\_model\_sasst(:,2),'b')

plot(chao\_delay\_model\_sasst(:,2),'k')

title('Total Tropospheric Delay using Sasstamoinen Model at 15\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(11)

hold on, grid on

plot(flat\_delay\_model\_hop(:,2),'r')

plot(misra\_delay\_model\_hop(:,2),'b')

plot(chao\_delay\_model\_hop(:,2),'k')

title('Total Tropospheric Delay using Hopfield Model at 15\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(12)

hold on, grid on

plot(flat\_delay\_model\_sasst(:,3),'r')

plot(misra\_delay\_model\_sasst(:,3),'b')

plot(chao\_delay\_model\_sasst(:,3),'k')

title('Total Tropospheric Delay using Sasstamoinen Model at 45\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(13)

hold on, grid on

plot(flat\_delay\_model\_hop(:,3),'r')

plot(misra\_delay\_model\_hop(:,3),'b')

plot(chao\_delay\_model\_hop(:,3),'k')

title('Total Tropospheric Delay using Hopfield Model at 45\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(14)

hold on, grid on

plot(flat\_delay\_model\_sasst(:,4),'r')

plot(misra\_delay\_model\_sasst(:,4),'b')

plot(chao\_delay\_model\_sasst(:,4),'k')

title('Total Tropospheric Delay using Sasstamoinen Model at 85\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')

figure(15)

hold on, grid on

plot(flat\_delay\_model\_hop(:,4),'r')

plot(misra\_delay\_model\_hop(:,4),'b')

plot(chao\_delay\_model\_hop(:,4),'k')

title('Total Tropospheric Delay using Hopfield Model at 85\circ'),xlabel('Time (sec)'),ylabel('Total Trophospheric Delay (m)')

legend('Flat Earth Mapping','Misra and Enge Mapping','Chao Mapping')