Lab 3 - Computation of receiver position from code and phase measurements

Kevin Mead

May 3, 2014

1 Introduction

The task is to compute the coordinates of receiver ROV at given time using P1-code and L1 phase observations from two GPS receivers: REF and ROV. The receiver REF has known coordinates. The calculations were done with observation files and other values given in the Excel document *DiffAssignmentL1.xls*. Unlike the last lab, where the satellite and receiver clocks were not synchronized, this lab synchronizes clocks of all the satellites to the system time.

2 Methodology

Matlab was used to solve for the following parameters and all of the following equations were taken from the technical report: Relative positioning with GPS.[1]

The steps and equations from this algorithm are documented in the Matlab code, Section 5. The pseudo ranges and phase observations for REF and ROV are given from the beginning in this lab. Two GPS receivers measured the pseudoranges to each satellite at the same time. Of the two points, we used point A as REF and B as ROV. The first step is to synchronize all the clocks for each satellite so we can combine the results. The equation used to synchronize the clocks,

$$\begin{aligned} P_A^s &= P_A^s(t_A) = P_A^s(\tilde{t}_A) + \dot{\rho}_A^s(\tilde{t}_A)\delta\hat{t}_A \\ \lambda\varphi_A^s &= \lambda\varphi_A^s(t_A) = \lambda\varphi_A^s(\tilde{t}_A) + \dot{\rho}_A^s(\tilde{t}_A)\delta\hat{t}_A \end{aligned}$$

where A is REF or ROV. Next, the ionospheric and tropospheric effects are eliminated common to both stations. The single differences are,

$$\begin{aligned} \lambda\varphi_{AB}^s &= \lambda\varphi_B^s - \lambda\varphi_A^s = \rho_{AB}^s + c\delta t_{AB} + \lambda N_{AB}^s \\ P_{AB}^s &= P_B^s - P_A^s = \rho_{AB}^s + c\delta t_{AB} \end{aligned}$$

which are then taken to form double differences and leads to the most important feature of this method: cancellation of the receiver clock errors. The double difference equations are computed by choosing a reference satellite and subtracting the phase and code observations of each satellite from it. Satellite 20 was chosen to be the reference in this lab. Double differences from reference satellite 20 is given from,

$$\lambda\varphi_{AB}^{st} = \lambda\varphi_B^t - \lambda\varphi_A^s = \rho_{AB}^{st} + \lambda N_{AB}^{st} \tag{1}$$

$$P_{AB}^{st} = P_B^t - P_A^s = \rho_{AB}^{st} \tag{2}$$

Linearizing Equation (1) and (2) and converting to matrix form, we get

$$
\begin{aligned}
L &= Ax \\
L &= \begin{bmatrix}
\lambda\varphi_{AB}^{20,4} - \rho_{AB,0}^{20,4} \\
\lambda\varphi_{AB}^{20,5} - \rho_{AB,0}^{20,5} \\
\lambda\varphi_{AB}^{20,6} - \rho_{AB,0}^{20,6} \\
\lambda\varphi_{AB}^{20,24} - \rho_{AB,0}^{20,24} \\
\lambda\varphi_{AB}^{20,25} - \rho_{AB,0}^{20,25} \\
P_{AB,1}^{20,4} - \rho_{AB,0}^{20,4} \\
P_{AB,1}^{20,5} - \rho_{AB,0}^{20,5} \\
P_{AB,1}^{20,6} - \rho_{AB,0}^{20,6} \\
P_{AB,1}^{20,24} - \rho_{AB,0}^{20,24} \\
P_{AB,1}^{20,25} - \rho_{AB,0}^{20,25}
\end{bmatrix} \\
A &= \begin{bmatrix}
a_{XB}^{20,4} & a_{YB}^{20,4} & a_{ZB}^{20,4} & \lambda_1 & 0 & 0 & 0 & 0 \\
a_{XB}^{20,5} & a_{YB}^{20,5} & a_{ZB}^{20,5} & 0 & \lambda_1 & 0 & 0 & 0 \\
a_{XB}^{20,6} & a_{YB}^{20,6} & a_{ZB}^{20,6} & 0 & 0 & \lambda_1 & 0 & 0 \\
a_{XB}^{20,24} & a_{YB}^{20,24} & a_{ZB}^{20,24} & 0 & 0 & 0 & \lambda_1 & 0 \\
a_{XB}^{20,25} & a_{YB}^{20,25} & a_{ZB}^{20,25} & 0 & 0 & 0 & 0 & \lambda_1 \\
a_{XB}^{20,4} & a_{YB}^{20,4} & a_{ZB}^{20,4} & 0 & 0 & 0 & 0 & 0 \\
a_{XB}^{20,5} & a_{YB}^{20,5} & a_{ZB}^{20,5} & 0 & 0 & 0 & 0 & 0 \\
a_{XB}^{20,6} & a_{YB}^{20,6} & a_{ZB}^{20,6} & 0 & 0 & 0 & 0 & 0 \\
a_{XB}^{20,24} & a_{YB}^{20,24} & a_{ZB}^{20,24} & 0 & 0 & 0 & 0 & 0 \\
a_{XB}^{20,25} & a_{YB}^{20,25} & a_{ZB}^{20,25} & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \\
x &= [\Delta X \quad \Delta Y \quad \Delta Z \quad N_{AB}^{20,4} \quad \cdots \quad N_{AB}^{20,4}]^{\mathrm{T}}
\end{aligned}
$$

The LSQ of these equations is given by

$$X = (A^T PA)^{-1} A^T PL \tag{3}$$

The coefficients $a_x,\ a_y,\ a_z,\ \rho_{AB,0}^{pq}$ are from the linearization of the topocentric distance differentiation,

$$
\begin{aligned}
a_x^s &= -\frac{X^s - X_{B0}}{\rho_{BO}^s} \\
a_y^s &= -\frac{Y^s - Y_{B0}}{\rho_{BO}^s} \\
a_z^s &= -\frac{Z^s - Z_{B0}}{\rho_{BO}^s}
\end{aligned}
$$

Similar to the phase and code double differencing, the coefficients are also subtracted differenced with the reference satellites values, in this case, Satellite 20. In Equation (3), the P matrix is the weight matrix and it is given in the excel file. The P matrix contains the standard deviation of the undifferenced phase and code, which for most receivers is 2 mm and 0.3 m respectively.

3 Results

The results for the LSQ of equations is,

$$\mathbf{X}(m) = \begin{bmatrix} \text{-0.03} \\ 0.16 \\ 0.22 \\ 951566.01 \\ 1014451.16 \\ \text{-1138240.42} \\ \text{-9347.42} \\ \text{-1078921.56} \end{bmatrix}$$

$$\mathbf{L}(m) = \begin{bmatrix} 181077.04 \\ 193043.66 \\ \text{-216600.08} \\ \text{-1778.67} \\ \text{-205312.07} \\ 0.13 \\ \text{-0.22} \\ \text{-0.09} \end{bmatrix}$$

$$\lambda\varphi = \begin{bmatrix} 180848.77 \\ 192567.29 \\ \text{-216415.03} \\ \text{-2197.22} \\ \text{-204701.52} \end{bmatrix}$$

$$P = \begin{bmatrix} \text{-228.14} \\ \text{-476.60} \\ 184.96 \\ \text{-418.71} \\ 610.13 \end{bmatrix}$$

. The results agree completely with the reference material given because both calculations used all of the 11 satellites available. The PDOP value was calculated using,

$$PDOP = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}$$

4 Analysis and Discussion

- Are the results reasonable? Compare the results with your expectations.

  Yes the results are reasonable. The final position of the receiver is very close to the approximate position and this is shown in Table **??**. We also get very consistent results if we compute the same calculations for different epochs.

Item		
Animal	Description	Price ($)
Gnat	per gram	13.65
	each	0.01
Gnu	stuffed	92.50
Emu	stuffed	33.33
Armadillo	frozen	8.99

- Can we draw any conclusion/implications from the results?

  We can conclude that the approximate coordinates are very close to the true receiver position, even though we did not factor in ionospheric and tropospheric effects.

- Are results reliable and accurate?

  I believe the results are reliable and accurate because the σ value is very small for all x,y,z and all 11 satellites were taken into account. But the least squares method can give poor results if there are any outliers or the errors aren't Gaussian distributed.[2] Looking at other epochs in the SPP Results, we can see that they also give approximately the same position for the receiver.

- Would it have been more appropriate to use another method? Does the method need to be further developed?

  I think it is only necessary to further develop this method if you needed a higher degree of accuracy. This method could be further developed to take into account the ionospheric and tropospheric effects and factor in all the given epochs for the satellites. I think it could me more appropriate to use the difference method is best used when you have more satellites and tropospheric and ionospheric effects are cancelled out in the equations. This means our final solution should be more accurate using the difference method.

References

[1] M. Horemuž, "Relative positioning with gps," Royal Institute of Technology, Tech. Rep., 2014.

[2] A. L. Garcia, *Numerical Methods for Physics (2Nd Edition)*, 2nd ed., A. Reeves and P. F. Corey, Eds. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1999.

5 Main Code

Contents

- 3. Compute coefficients ax , ay , az and rhoAB_0
- 4. Fill in matrixes A, L and compute least square solution of

Lab 3

Difference Equations for GPS

```
clear all; clc;
```

Import file

```
addpath(['/Users/kevin/SkyDrive/KTH Work/'...
    'Period 4 2014/GNSS/Labs/'...
    'L3 - Computation of receiver position '...
    'from code and phase measurements'])
excfile = xlsread('DiffAssignmentL1');
excfile = padarray(excfile,5,'pre');
cellDiff = num2cell(excfile); % same as a cell
```

Constants

```
c = 299792458; % speed of light (m/s)
[lambda1,lambda2] = cellDiff{78:79,2};
mu = 3.986005e14; % universal gravitational parameter (m/s)^3
omega_e_dot = 7.2921151467e-5; % earth rotation rate (rad/s)
F = -4.442807633e-10; % s/m^1/2
nRows = 5;
```

Variables

```
dtA = excfile(24,3);
dtB = excfile(25,3);
sNum = excfile((28:33),1);
refSatNum = 20;
for i = 1:length(sNum) % vector of ones at satellite numbers
    vectOnes(sNum(i),1) = 1;
end
[Xref,Xrov,Yref,Yrov,Zref,Zrov] = cellDiff{(37:38),3:5};
for i = 43:48 %
    rhoDot(excfile(i-15,1),1) = excfile(i-15,2);
    satCoordinates(excfile(i,1),:) = excfile(i,3:5);
    rhoAtoP(excfile(i+23,1),:) = excfile(i+23,2);
    rhoB0toP(excfile(i+23,1),:) = excfile(i+23,3);
end
count = 1;
for i = 6:3:21
   P1ref(sNum(count),1) = excfile(i,3);
   P1rov(sNum(count),1) = excfile(i+1,3);
   phi1ref(sNum(count),1) = excfile(i,4);
   phi1rov(sNum(count),1) = excfile(i+1,4);
```

```
   count = count + 1;
end
weightMatrix = excfile(52:61,1:10);
```

Main steps of Calculation

1. Synchronize observables from both receivers using equations (14)

```
P1refeq14 = P1ref + rhoDot*dtA;
P1roveq14 = P1rov + rhoDot*dtB;
phi1refeq14 = lambda1*phi1ref + rhoDot * dtA;
phi1roveq14 = lambda1*phi1rov + rhoDot * dtB;
```

2. Compute single and double differences - equations (15) and (18). Use satellite 20 as a reference satellite for double differencing.

```
P1eq15 = P1refeq14 - P1roveq14;
phi1eq15 = (phi1refeq14 - phi1roveq14);
dtAB = dtB - dtA;
% Double differences
phi1eq18 = phi1eq15 - phi1eq15(refSatNum)*vectOnes;
P1eq18 = P1eq15 - P1eq15(refSatNum)*vectOnes;
```

3. Compute coefficients a_X , a_Y , a_Z and $rho_{AB,0}^{pq}$ - equations (12).

```
ax_S_rov = -1 * (satCoordinates(:,1)-Xrov*vectOnes)./(rhoB0toP);
ay_S_rov = -1 * (satCoordinates(:,2)-Yrov*vectOnes)./(rhoB0toP);
az_S_rov = -1 * (satCoordinates(:,3)-Zrov*vectOnes)./(rhoB0toP);
aXB_st = -ax_S_rov + ax_S_rov(20);
aYB_st = -ay_S_rov + ay_S_rov(20);
aZB_st = -az_S_rov + az_S_rov(20);
rhoAB_0 = rhoAtoP - rhoB0toP;
```

4. Fill in matrixes A, L and compute least square solution of

equations (21).

```
p_AB_s = rhoB0toP-rhoAtoP;
nv = [4,5,6,24,25];
A = [aXB_st(nv,1), aYB_st(nv,1), aZB_st(nv,1)];
A = [ A , eye(length(nv)) * lambda1 ; A , zeros(length(nv)) ];
% L
L = [phi1eq18(nv) - rhoAB_0(nv) + ones(5,1)*rhoAB_0(20);...
    P1eq18(nv) - rhoAB_0(nv) + ones(5,1)*rhoAB_0(20)];
%
X = inv(A'*weightMatrix*A)*A'*weightMatrix*L;
```