

APPENDIX A

LINEARIZING THE GPS PSEUDORANGE EQUATIONS

In this appendix we show that all but one column of the H matrix are direction cosines from the receiver to each of the satellites. The pseudorange equations we wish to linearize are as described in Eq. 2.10, but with a modification on the left hand side:

$$\begin{aligned} PR_1 - \Delta t_1 &= \sqrt{(X-X_1)^2 + (Y-Y_1)^2 + (Z-Z_1)^2} + ct_B \\ PR_2 - \Delta t_2 &= \sqrt{(X-X_2)^2 + (Y-Y_2)^2 + (Z-Z_2)^2} + ct_B \\ PR_3 - \Delta t_3 &= \sqrt{(X-X_3)^2 + (Y-Y_3)^2 + (Z-Z_3)^2} + ct_B \\ PR_4 - \Delta t_4 &= \sqrt{(X-X_4)^2 + (Y-Y_4)^2 + (Z-Z_4)^2} + ct_B \end{aligned} \tag{A.1}$$

where the Δt_i terms represent satellite clock corrections. GPS operates on the principle that all of the satellites are synchronized with GPS master time. The pseudorange measurements must share a common time basis or the position solution will be highly inaccurate. The satellite clocks, however, are not in synchronization with one another. Therefore, the Air Force monitors each satellite and regularly uploads clock correction parameters which can be applied by the user. The satellite clock correction is given by Eq. 7.14, which is repeated here:

$$\Delta t_{SV} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_{per} \tag{A.2}$$

A further correction is made for single frequency users because the a_{f0} clock offset term is based on dual frequency observations:

$$(\Delta t_{SV})_{L1} = \Delta t_{SV} - \tau_{GD} \quad (\text{A.3})$$

where τ_{GD} is a group delay differential given by:

$$\tau_{GD} = \frac{1}{1 - \left(\frac{f_{L1}}{f_{L2}} \right)^2} (t_{L1} - t_{L2}) \quad (\text{A.4})$$

which is a quantity measured on the ground before satellite launch. It should be noted that the mean value of τ_{GD} can be any value in the range ± 15 ns with random variations of 3 ns (2σ). Thus, τ_{GD} could have a magnitude of 18 ns or larger for a given satellite.

We consider four pseudoranges here because that is the minimum required to solve for the four unknowns of three dimensional position and receiver clock offset. A Taylor Series is used to expand these equations about an estimated position $(\hat{X}, \hat{Y}, \hat{Z}, \hat{t}_B)$ with the higher order terms ignored. For pseudorange measurement i :

$$\begin{aligned} PR_i - \Delta t_i \approx & P\hat{R}_i + \frac{\partial PR_i}{\partial \hat{X}} \Big|_{(\hat{X}, \hat{Y}, \hat{Z})} \partial X + \frac{\partial PR_i}{\partial \hat{Y}} \Big|_{(\hat{X}, \hat{Y}, \hat{Z})} \partial Y \\ & + \frac{\partial PR_i}{\partial \hat{Z}} \Big|_{(\hat{X}, \hat{Y}, \hat{Z})} \partial Z + c\hat{t}_B \end{aligned} \quad (\text{A.5})$$

where $(P\hat{R}_i - \Delta t_i)$ is the pseudorange based on the estimated position. Thus, each pseudorange can be approximated as:

$$PR_i - \Delta t_i \approx P\hat{R}_i + \frac{(\hat{X}-X_i)\partial X + (\hat{Y}-Y_i)\partial Y + (\hat{Z}-Z_i)\partial Z}{\sqrt{(\hat{X}-X_i)^2 + (\hat{Y}-Y_i)^2 + (\hat{Z}-Z_i)^2}} + ct_B \quad (\text{A.6})$$

Now we use the following substitutions:

$$\partial PR_i = PR_i - \Delta t_i - P\hat{R}_i \quad (\text{A.7})$$

$$\hat{R}_i = \sqrt{(\hat{X}-X_i)^2 + (\hat{Y}-Y_i)^2 + (\hat{Z}-Z_i)^2}$$

The result is:

$$\partial PR_i = \frac{\hat{X}-X_i}{\hat{R}_i}\partial X + \frac{\hat{Y}-Y_i}{\hat{R}_i}\partial Y + \frac{\hat{Z}-Z_i}{\hat{R}_i}\partial Z + ct_B \quad (\text{A.8})$$

The coefficients of ∂X , ∂Y , and ∂Z are direction cosines. We can now rewrite the linearized equations of A.1 in matrix form:

$$\begin{pmatrix} \partial PR_1 \\ \partial PR_2 \\ \partial PR_3 \\ \partial PR_4 \end{pmatrix} = \begin{bmatrix} \frac{\hat{X}-X_1}{\hat{R}_1} & \frac{\hat{Y}-Y_1}{\hat{R}_1} & \frac{\hat{Z}-Z_1}{\hat{R}_1} & 1 \\ \frac{\hat{X}-X_2}{\hat{R}_2} & \frac{\hat{Y}-Y_2}{\hat{R}_2} & \frac{\hat{Z}-Z_2}{\hat{R}_2} & 1 \\ \frac{\hat{X}-X_3}{\hat{R}_3} & \frac{\hat{Y}-Y_3}{\hat{R}_3} & \frac{\hat{Z}-Z_3}{\hat{R}_3} & 1 \\ \frac{\hat{X}-X_4}{\hat{R}_4} & \frac{\hat{Y}-Y_4}{\hat{R}_4} & \frac{\hat{Z}-Z_4}{\hat{R}_4} & 1 \end{bmatrix} \begin{pmatrix} \partial X \\ \partial Y \\ \partial Z \\ ct_B \end{pmatrix} \quad (\text{A.9})$$

Equation A.9 can be expressed as follows:

$$\partial \underline{\mathbf{Y}} = \mathbf{H} \partial \underline{\mathbf{\beta}} \quad (\text{A.10})$$

where these substitutions have been made:

$$\partial \underline{\mathbf{Y}} \triangleq \begin{pmatrix} \partial PR_1 \\ \partial PR_2 \\ \partial PR_3 \\ \partial PR_4 \end{pmatrix} \quad \mathbf{H} \triangleq \begin{bmatrix} \frac{\hat{X}-X_1}{\hat{R}_1} & \frac{\hat{Y}-Y_1}{\hat{R}_1} & \frac{\hat{Z}-Z_1}{\hat{R}_1} & 1 \\ \frac{\hat{X}-X_2}{\hat{R}_2} & \frac{\hat{Y}-Y_2}{\hat{R}_2} & \frac{\hat{Z}-Z_2}{\hat{R}_2} & 1 \\ \frac{\hat{X}-X_3}{\hat{R}_3} & \frac{\hat{Y}-Y_3}{\hat{R}_3} & \frac{\hat{Z}-Z_3}{\hat{R}_3} & 1 \\ \frac{\hat{X}-X_4}{\hat{R}_4} & \frac{\hat{Y}-Y_4}{\hat{R}_4} & \frac{\hat{Z}-Z_4}{\hat{R}_4} & 1 \end{bmatrix} \quad \partial \underline{\mathbf{\beta}} \triangleq \begin{pmatrix} \partial X \\ \partial Y \\ \partial Z \\ ct_B \end{pmatrix} \quad (\text{A.11})$$

An iterative computation is carried out to converge on the position solution. In general, $\partial \underline{\mathbf{Y}}$ is an $m \times 1$ vector where m is the number of satellites in view. Accordingly, \mathbf{H} is an $m \times 4$ matrix.

One method of solving A.10 is to take a generalized inverse of \mathbf{H} :

$$\partial \underline{\mathbf{\beta}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \partial \underline{\mathbf{Y}} \quad (\text{A.12})$$

The result can be used to converge on the solution within four or five iterations [Diggle, 1994].

It is not necessary for the initial estimated position and receiver clock offset to be accurate. The basic algorithm is as follows:

Calculate satellite positions
Apply satellite clock corrections to the pseudorange measurements
Form initial position estimate
Iterate until convergence,
 Calculate approximate pseudoranges based on position estimate and SV positions
 Form the geometry matrix H
 Subtract measured pseudoranges from estimated pseudoranges
 Update the user state by solving Eq. A.9
End Loop