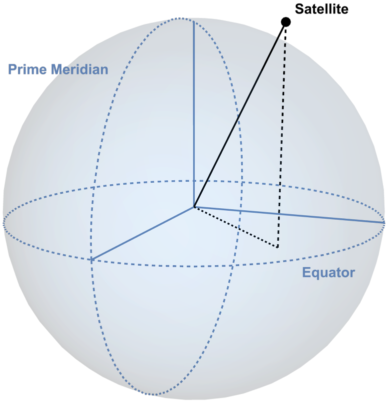
Part 8: Solving

Building a GPS receiver from scratch

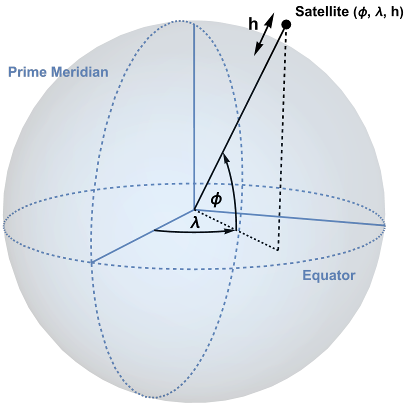
Chris Doble

Topics

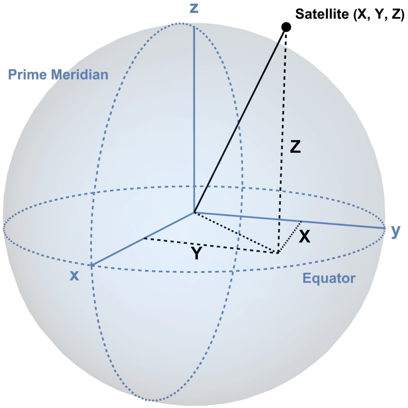
Coordinate system



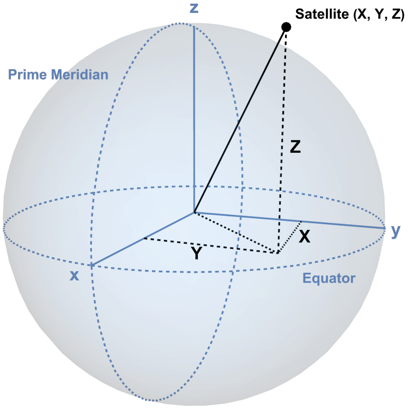
Coordinate system



Coordinate system

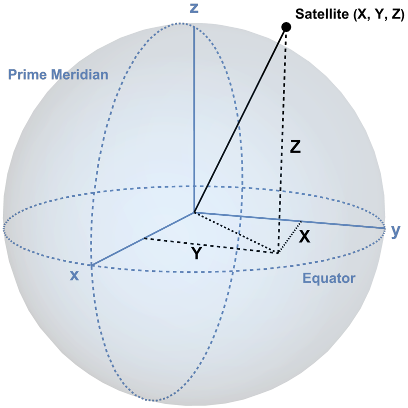


Coordinate system



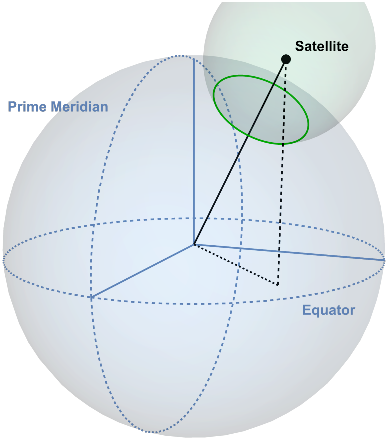
Transit time T

Coordinate system



Transit time $T \Rightarrow$ distance cT

Coordinate system



Transit time $T \Rightarrow$ distance cT

The pseudorange equation

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}=cT$$

where (x, y, z) is our unknown location.

The pseudorange equation

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-cT=0$$

where (x, y, z) is our unknown location.

The pseudorange equation

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

where (x, y, z) is our unknown location and t is our clock bias.

The pseudorange equation

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

where (x, y, z) is our unknown location and t is our clock bias.

The pseudorange equation

$$\sqrt{(X - x)^2 + (Y - y)^2 + (Z - z)^2} - c(T - t) = 0$$

where (x, y, z) is our unknown location and t is our clock bias.

Topics

1. The pseudorange equation
2. Satellite location and transit time
3. System of equations
   - Definition
   - Solving

Transmission time

Transmission time

The user shall correct the time received from the SV with the equation (in seconds)

$$t = t_{sv} - \Delta t_{sv} \qquad (1)$$

where

t = GPS system time (seconds),

t_{sv} = effective SV PRN code phase time at message transmission time (seconds),

Δt_{sv} = SV PRN code phase time offset (seconds).

The SV PRN code phase offset is given by

$$\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_r \qquad (2)$$

$$\vdots$$

Thus, the user who utilizes the L1 C/A signal only shall modify the code phase offset in accordance with paragraph 20.3.3.3.3.1 with the equation

$$(\Delta t_{SV})_{L1C/A} = \Delta t_{SV} - T_{GD}$$

Transmission time

The user shall correct the time received from the SV with the equation (in seconds)

$$t = \boxed{t_{sv}} - \Delta t_{sv} \qquad (1)$$

where

t = GPS system time (seconds),

$\boxed{t_{sv} = \text{effective SV PRN code phase time at message transmission time (seconds),}}$

Δt_{sv} = SV PRN code phase time offset (seconds).

The SV PRN code phase offset is given by

$$\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_r \qquad (2)$$

⋮

Thus, the user who utilizes the L1 C/A signal only shall modify the code phase offset in accordance with paragraph 20.3.3.3.3.1 with the equation

$$(\Delta t_{SV})_{L1C/A} = \Delta t_{SV} - T_{GD}$$

Transmission time

The user shall correct the time received from the SV with the equation (in seconds)

$$t = t_{sv} - \boxed{\Delta t_{sv}} \qquad (1)$$

where

t = GPS system time (seconds),

t_{sv} = effective SV PRN code phase time at message transmission time (seconds),

$\boxed{\Delta t_{sv} = \text{SV PRN code phase time offset (seconds).}}$

The SV PRN code phase offset is given by

$$\boxed{\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_r} \qquad (2)$$

$$\vdots$$

Thus, the user who utilizes the L1 C/A signal only shall modify the code phase offset in accordance with paragraph 20.3.3.3.3.1 with the equation

$$(\Delta t_{SV})_{L1C/A} = \Delta t_{SV} - T_{GD}$$

Transmission time

The user shall correct the time received from the SV with the equation (in seconds)

$$t = t_{sv} - \Delta t_{sv} \qquad (1)$$

where

t = GPS system time (seconds),

t_{sv} = effective SV PRN code phase time at message transmission time (seconds),

Δt_{sv} = SV PRN code phase time offset (seconds).

The SV PRN code phase offset is given by

$$\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_r \qquad (2)$$

$$\vdots$$

Thus, the user who utilizes the L1 C/A signal only shall modify the code phase offset in accordance with paragraph 20.3.3.3.3.1 with the equation

$$\boxed{(\Delta t_{SV})_{L1C/A} = \Delta t_{SV} - T_{GD}}$$

Transmission time

The user shall correct the time received from the SV with the equation (in seconds)

$$t = \boxed{t_{sv}} - \Delta t_{sv} \qquad (1)$$

where

t = GPS system time (seconds),

$\boxed{t_{sv} = \text{effective SV PRN code phase time at message transmission time (seconds),}}$

Δt_{sv} = SV PRN code phase time offset (seconds).

The SV PRN code phase offset is given by

$$\Delta t_{sv} = a_{f0} + a_{f1}(t - t_{oc}) + a_{f2}(t - t_{oc})^2 + \Delta t_r \qquad (2)$$

⋮

Thus, the user who utilizes the L1 C/A signal only shall modify the code phase offset in accordance with paragraph 20.3.3.3.3.1 with the equation

$$(\Delta t_{SV})_{L1C/A} = \Delta t_{SV} - T_{GD}$$

Transmission time

$$t_{sv} = \mathrm{TOW} \times 6\,\mathrm{s}$$

Transmission time

$$t_{sv} = \text{TOW} \times 6\,\text{s} + \text{PRN count} \times 1\,\text{ms}$$

Reception time

Reception time

1. Record the time at which we finish receiving the PRN code (our clock)

Reception time

1. Record the time at which we finish receiving the PRN code (our clock)
2. Add 18 leap seconds

Reception time

1. Record the time at which we finish receiving the PRN code (our clock)
2. Add 18 leap seconds
3. Calculate the number of seconds since GPS started operating

Reception time

1. Record the time at which we finish receiving the PRN code (our clock)
2. Add 18 leap seconds
3. Calculate the number of seconds since GPS started operating
4. Calculate the remainder when divided by the number of seconds in a GPS week

Location

$\mu = 3.986005 \text{ x } 10^{14}$ meters^3/sec^2

$\dot{\Omega}_e = 7.2921151467 \text{ x } 10^{-5}$ rad/sec

$A = \left(\sqrt{A}\right)^2$

$n_0 = \sqrt{\frac{\mu}{A^3}}$

$t_k = t - t_{oe}$*

$n = n_0 + \Delta n$

$M_k = M_0 + nt_k$

$E_0 = M_k$

$$E_j = E_{j-1} + \frac{M_k - E_{j-1} + e \sin E_{j-1}}{1 - e \cos E_{j-1}}$$

$E_k = E_j$

$$\nu_k = 2 \tan^{-1}\left(\sqrt{\frac{1+e}{1-e}} \tan \frac{E_k}{2}\right)$$

$\Phi_k = \nu_k + \omega$

$\delta u_k = c_{us}\sin 2\Phi_k + c_{uc}\cos 2\Phi_k$
$\delta r_k = c_{rs}\sin 2\Phi_k + c_{rc}\cos 2\Phi_k$
$\delta i_k = c_{is}\sin 2\Phi_k + c_{ic}\cos 2\Phi_k$

$u_k = \Phi_k + \delta u_k$

$r_k = A(1 - e \cos E_k) + \delta r_k$

$i_k = i_0 + \delta i_k + (IDOT)\, t_k$

$x_k' = r_k \cos u_k$
$y_k' = r_k \sin u_k$

$\Omega_k = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_e)\, t_k - \dot{\Omega}_e\, t_{oe}$

$x_k = x_k' \cos\Omega_k - y_k' \cos i_k \sin\Omega_k$
$y_k = x_k' \sin\Omega_k + y_k' \cos i_k \cos\Omega_k$
$z_k = y_k' \sin i_k$

$\mu = 3.986005 \times 10^{14}$ meters^3/sec^2

$\dot{\Omega}_e = 7.2921151467 \times 10^{-5}$ rad/sec

$A = \left(\sqrt{A}\right)^2$

$n_0 = \sqrt{\frac{\mu}{A^3}}$

$t_k = \boxed{t} - t_{oe}$*

$n = n_0 + \Delta n$

$M_k = M_0 + nt_k$

$E_0 = M_k$

$E_j = E_{j-1} + \frac{M_k - E_{j-1} + e \sin E_{j-1}}{1 - e \cos E_{j-1}}$

$E_k = E_j$

$\nu_k = 2 \tan^{-1}\left(\sqrt{\frac{1+e}{1-e}} \tan \frac{E_k}{2}\right)$

$\Phi_k = \nu_k + \omega$

$\delta u_k = c_{us}\sin 2\Phi_k + c_{uc}\cos 2\Phi_k$
$\delta r_k = c_{rs}\sin 2\Phi_k + c_{rc}\cos 2\Phi_k$
$\delta i_k = c_{is}\sin 2\Phi_k + c_{ic}\cos 2\Phi_k$

$u_k = \Phi_k + \delta u_k$

$r_k = A(1 - e \cos E_k) + \delta r_k$

$i_k = i_0 + \delta i_k + (IDOT)\, t_k$

$x_k' = r_k \cos u_k$
$y_k' = r_k \sin u_k$

$\Omega_k = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_e)\, t_k - \dot{\Omega}_e\, t_{oe}$

$x_k = x_k' \cos\Omega_k - y_k' \cos i_k \sin\Omega_k$
$y_k = x_k' \sin\Omega_k + y_k' \cos i_k \cos\Omega_k$
$z_k = y_k' \sin i_k$

The pseudorange equation

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

Topics

Topics

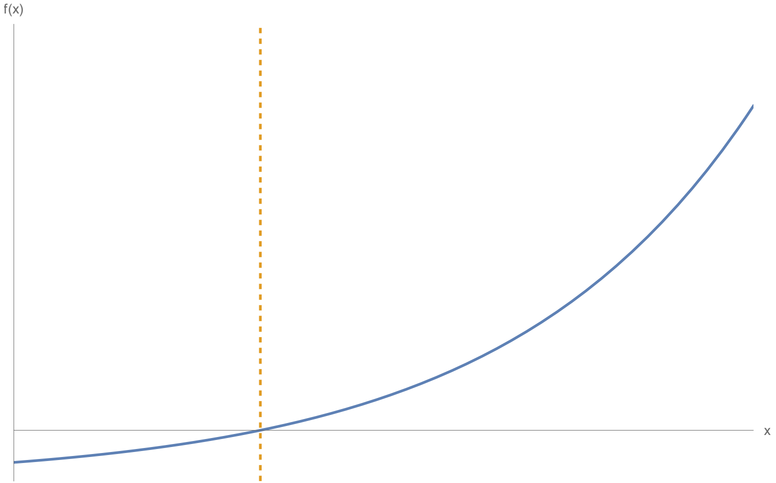
Definition

$$\sqrt{(X_1 - x)^2 + (Y_1 - y)^2 + (Z_1 - z)^2} - c(T_1 - t) = 0$$
$$\sqrt{(X_2 - x)^2 + (Y_2 - y)^2 + (Z_2 - z)^2} - c(T_2 - t) = 0$$
$$\sqrt{(X_3 - x)^2 + (Y_3 - y)^2 + (Z_3 - z)^2} - c(T_3 - t) = 0$$
$$\sqrt{(X_4 - x)^2 + (Y_4 - y)^2 + (Z_4 - z)^2} - c(T_4 - t) = 0$$

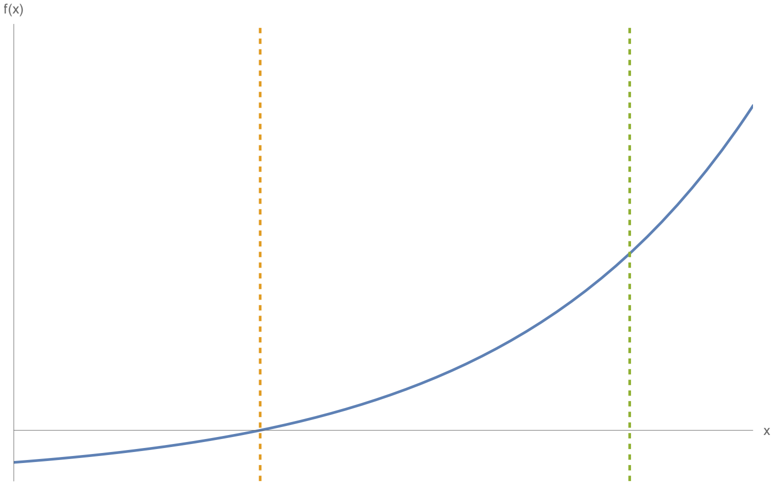
Topics

The Newton-Raphson method

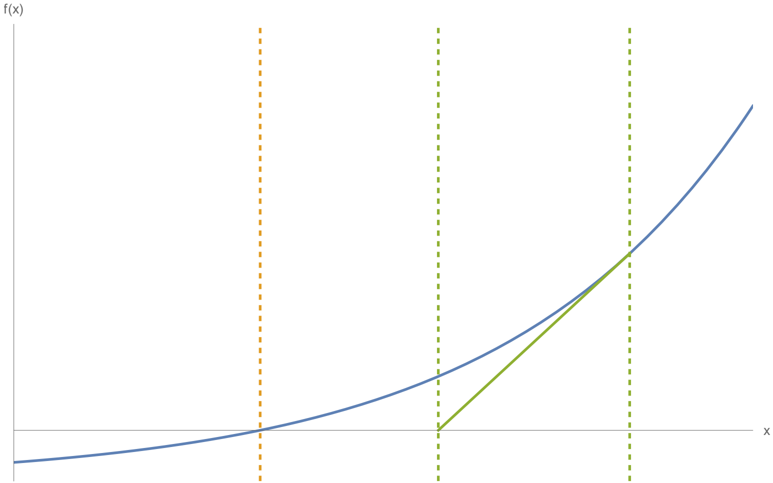
The Newton-Raphson method



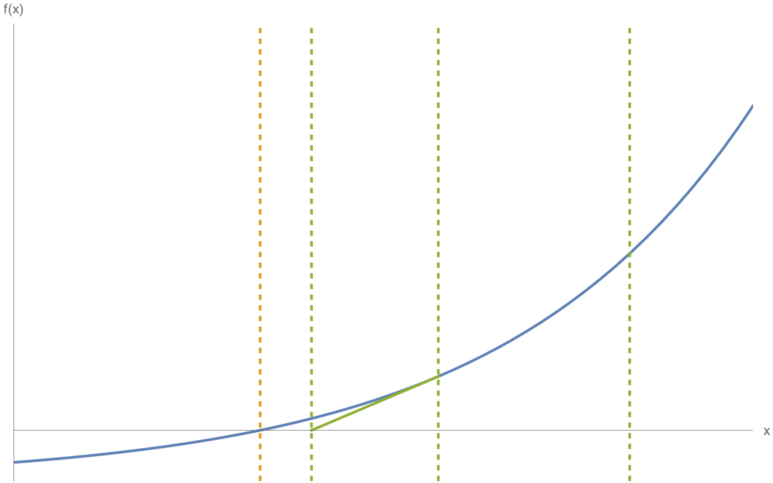
The Newton-Raphson method



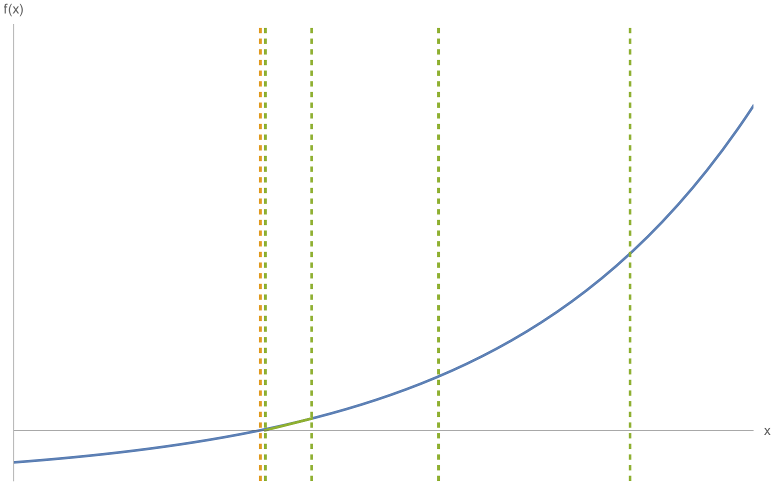
The Newton-Raphson method



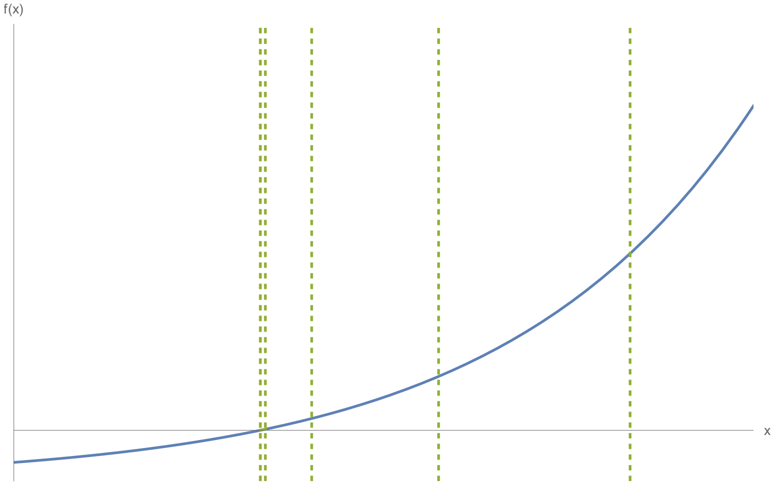
The Newton-Raphson method



The Newton-Raphson method



The Newton-Raphson method



The Gauss-Newton algorithm

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero
- Initial guess $\beta_0 = (x_0, y_0, z_0, t_0)$

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero
- Initial guess $\beta_0 = (x_0, y_0, z_0, t_0)$
- The Jacobian matrix J is the multi-dimensional equivalent of the derivative

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero
- Initial guess $\beta_0 = (x_0, y_0, z_0, t_0)$
- The Jacobian matrix J is the multi-dimensional equivalent of the derivative
- Cell i, j tells us how the result of equation i would change if we changed variable j

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero
- Initial guess $\beta_0 = (x_0, y_0, z_0, t_0)$
- The Jacobian matrix J is the multi-dimensional equivalent of the derivative
- Cell i, j tells us how the result of equation i would change if we changed variable j
- We iteratively improve the guess $\beta_{n+1} = \beta_n - (J^T J)^{-1} J^T r(\beta_n)$ where $r(\beta)$ is the residual vector (a column vector containing the results of evaluating the pseudorange equations)

The Gauss-Newton algorithm

- Tries to make our four (or more) pseudorange equations of four variables all equal zero
- Initial guess $\beta_0 = (x_0, y_0, z_0, t_0)$
- The Jacobian matrix J is the multi-dimensional equivalent of the derivative
- Cell i, j tells us how the result of equation i would change if we changed variable j
- We iteratively improve the guess $\beta_{n+1} = \beta_n - (J^T J)^{-1} J^T r(\beta_n)$ where $r(\beta)$ is the residual vector (a column vector containing the results of evaluating the pseudorange equations)
- Convert from ECEF to geodetic coordinates using Bowring's method

Hooray!

Recap

Recap

- We express locations using ECEF coordinates

Recap

- We express locations using ECEF coordinates
- The pseudorange equation relates our distance to the satellite and the signal transit time

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

Recap

- We express locations using ECEF coordinates
- The pseudorange equation relates our distance to the satellite and the signal transit time

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

- The signal transit time is defined as $T = t_{\text{received}} - t_{\text{transmitted}}$

Recap

- We express locations using ECEF coordinates
- The pseudorange equation relates our distance to the satellite and the signal transit time

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2} - c(T-t) = 0$$

- The signal transit time is defined as $T = t_{\text{received}} - t_{\text{transmitted}}$
- The satellite's location can be calculated using equations in the GPS spec

Recap

- We express locations using ECEF coordinates
- The pseudorange equation relates our distance to the satellite and the signal transit time

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

- The signal transit time is defined as $T = t_{\text{received}} - t_{\text{transmitted}}$
- The satellite's location can be calculated using equations in the GPS spec
- We need a system of at least four pseudorange equations to solve for x, y, z, and t

Recap

- We express locations using ECEF coordinates
- The pseudorange equation relates our distance to the satellite and the signal transit time

$$\sqrt{(X-x)^2+(Y-y)^2+(Z-z)^2}-c(T-t)=0$$

- The signal transit time is defined as $T = t_{\text{received}} - t_{\text{transmitted}}$
- The satellite's location can be calculated using equations in the GPS spec
- We need a system of at least four pseudorange equations to solve for x, y, z, and t
- The Gauss-Newton algorithm is used to estimate x, y, z, and t

Thank you!

https://github.com/chrisdoble/gps-receiver