Chapter 7
Differential Positioning and Carrier Phase Positioning

Single Point/Standalone Positioning

Review of Single Point Positioning (1/2)

- Single point positioning was originally presented in Chapter 2 and is briefly reviewed here for completeness. A single point position is when you compute your solution using pseudorange data from only one receiver. Expanding the measurement model for the i-th satellite gives

$$
\begin{aligned}
P_i &= \rho_i + cdt + \varepsilon_P \\
&= \sqrt{\left(x_i^s - x^r\right)^2 + \left(y_i^s - y^r\right)^2 + \left(z_i^s - z^r\right)^2} + cdt + \varepsilon_i^P \\
&= f\left(\hat{\mathbf{x}}\right)
\end{aligned}
$$

In this case, the unknowns are the position of the receiver (x^r,y^r,z^r) and the receiver clock bias in units of length (cdt):

$$
\mathbf{x}^T = \begin{bmatrix} x^r & y^r & z^r & cdt \end{bmatrix}
$$

Finally, the error term ε_P is the total of all errors (Chapter 6).

Review of Single Point Positioning (2/2)

- The measurement is obviously non-linear with respect to our unknowns and so we have to linearize the model. Taking the partial derivatives with respect to the unknowns gives the design matrix:

$$A_{SP,i} = \left.\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}} = \left.\begin{bmatrix} \frac{\partial P}{x^r} & \frac{\partial P}{y^r} & \frac{\partial P}{z^r} & \frac{\partial P}{cdt} \end{bmatrix}\right|_{\mathbf{x}=\hat{\mathbf{x}}} = \begin{bmatrix} \frac{\hat{x}^r - x_i^s}{\hat{\rho}_i} & \frac{\hat{y}^r - y_i^s}{\hat{\rho}_i} & \frac{\hat{z}^r - z_i^s}{\hat{\rho}_i} & 1 \end{bmatrix}$$

The weighted least-squares solution is given by

$$\delta\hat{\mathbf{x}} = \left(A_{SP}^T C_l^{-1} A_{SP}\right)^{-1} A_{SP}^T C_l^{-1}\mathbf{w} \;\Rightarrow\; \hat{\mathbf{x}} = \hat{\mathbf{x}} + \delta\hat{\mathbf{x}}$$

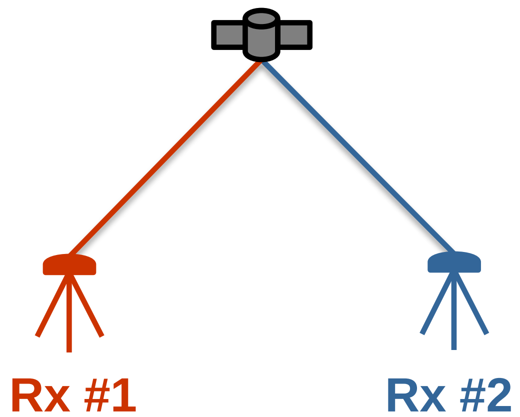
where C_l is the covariance matrix of the measurement errors

Single Differencing

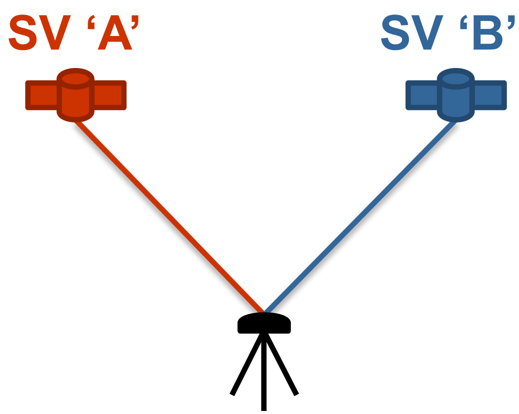
Single Differencing Overview

- The concept of single differencing was first presented at the end of the Chapter 5. To this end, we identified between-receiver and between-satellite single differences:

Between-Receiver Single Difference (Δ)



Between-Satellite Single Difference (∇)



- On the following slides, we revisit these concept in light of the knowledge of the various GNSS errors gained in Chapter 6. We also begin to look at using carrier phase data, instead of just the pseudorange.

Between-Receiver Single Differencing (1/2)

- The between-receiver single difference measurement equations can be written as

$$
\begin{aligned}
\Delta P_{12} &= P_2 - P_1 \\
&= \Delta\rho + \Delta d\rho + c(\Delta dT - \Delta dt) + \Delta d_{iono} + \Delta d_{trop} + \Delta m_P + \Delta n_P \\
\Delta\Phi_{12} &= \Phi_2 - \Phi_1 \\
&= \Delta\rho + \Delta d\rho + c(\Delta dT - \Delta dt) - \Delta d_{iono} + \Delta d_{trop} + \Delta m_\Phi + \Delta n_\Phi + \lambda\Delta N
\end{aligned}
$$

- What can we say about the errors?

Error	Magnitude
$\Delta d\rho$	
$c\Delta dT$	
$c\Delta dt$	
Δd_{iono}	
Δd_{trop}	

Error	Magnitude
Δm_P	
Δn_P	
Δm_Φ	
Δn_Φ	

Between-Receiver Single Differencing (2/2)

- The geometric range term can be written as

$$\begin{aligned}\Delta\rho &= \rho_2 - \rho_1 \\ &= \left|\vec{r}_{SV} - \vec{r}_{Rx2}\right| - \left|\vec{r}_{SV} - \vec{r}_{Rx1}\right|\end{aligned}$$

- Normally, it is assumed that receiver #1 is setup at a known location and this receiver is thus often called a “base station”.
- In this case, the position of the second receiver — the “rover” receiver — is computed relative to the base station.
  - In other words, we are now performing ***relative positioning***, not absolute positioning. This is also the origin of the term ***differential positioning***, as in ***Differential GPS/GNSS (DGPS/DGNSS)***

Between-Receiver Pseudorange

- If the base station position is known, and given our knowledge of the errors, the unknowns (to be estimated) include
  - Position of rover receiver (x,y,z or ϕ,λ,h)
  - Relative clock error (cΔdt)

- In this case, the design matrix is similar that of single point positioning

$$A_{\Delta P,i} = \left[\frac{\partial \Delta P}{x_{Rx2}} \quad \frac{\partial \Delta P}{y_{Rx2}} \quad \frac{\partial \Delta P}{z_{Rx2}} \quad \frac{\partial \Delta P}{c\Delta dt} \right]\Bigg|_{\mathbf{x}=\hat{\mathbf{x}}}$$

$$= \left[\frac{\left(\hat{x}_{Rx2} - x_i^s\right)}{\hat{\rho}_i^s} \quad \frac{\left(\hat{y}_{Rx2} - y_i^s\right)}{\hat{\rho}_i^s} \quad \frac{\left(\hat{z}_{Rx2} - z_i^s\right)}{\hat{\rho}_i^s} \quad -1 \right]$$

Between-Receiver Carrier Phase

- For carrier phase measurements, we also have to include the ambiguity terms. In this case, the unknowns are
  - Position of rover receiver (x,y,z or ϕ,λ,h)
  - Relative clock error (cΔdt)
  - Ambiguities (one per satellite; i.e., ΔN^i_{12} for the i-th satellite)

$$\vec{x} = \begin{bmatrix} x_{Rx2} & y_{Rx2} & z_{Rx2} & c\Delta dt & \Delta N^1_{12} & \Delta N^2_{12} & \cdots & \Delta N^M_{12} \end{bmatrix}^T$$

- The corresponding design matrix is

$$A_{\Delta\Phi,i} = \left. \begin{bmatrix} \frac{\partial \Delta\Phi}{x_{Rx2}} & \frac{\partial \Delta\Phi}{y_{Rx2}} & \frac{\partial \Delta\Phi}{z_{Rx2}} & \frac{\partial \Delta\Phi}{c\Delta dt} & \frac{\partial \Delta\Phi}{\Delta N^1_{12}} & \cdots & \frac{\partial \Delta\Phi}{\Delta N^i_{12}} & \cdots & \frac{\partial \Delta\Phi}{\Delta N^M_{12}} \end{bmatrix} \right|_{\mathbf{x}=\hat{\mathbf{x}}}$$

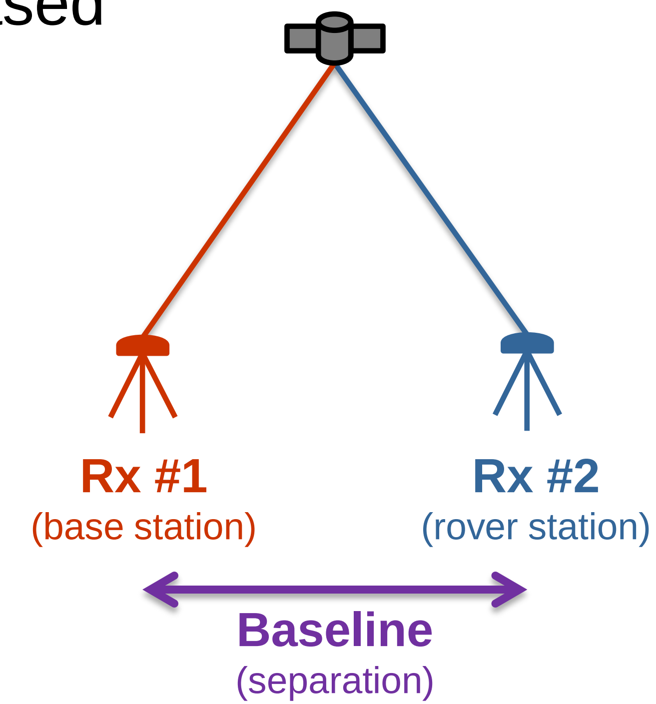
$$= \begin{bmatrix} \frac{\left(\hat{x}_{Rx2} - x^s_i\right)}{\hat{\rho}^s_i} & \frac{\left(\hat{y}_{Rx2} - y^s_i\right)}{\hat{\rho}^s_i} & \frac{\left(\hat{z}_{Rx2} - z^s_i\right)}{\hat{\rho}^s_i} & -1 & 0 & \cdots & \lambda & \cdots & 0 \end{bmatrix}$$

- What can we say about this matrix?

At the element of the i-th ambiguity (ΔN_{1i})

Summary of Between-Receiver Differencing

- The key benefits of between-receiver differencing are
  - Reduces errors correlated between the base and remote stations. The extent of the reduction is a function of the spatial decorrelation of the error source and the distance ("baseline") between the receivers.
  - Eliminates the satellite clock error (i.e., these are perfectly correlated).
- Main drawback is that the standard deviation of multipath and noise increased by $\sqrt{2}$ relative to the undifferenced (i.e., "raw") measurements.
- The estimated parameters include
  - Position of rover receiver (x,y,z or ϕ,λ,h)
  - Relative clock error ($c\Delta dt$)
  - Ambiguities (carrier phase case only)
- Cannot separate clock from ambiguity terms.



Alternate Between-Receiver Difference (1/2)

- The previous slides require that measurements from the base and rover stations are available at the same time in order to difference them.
  - This imposes constraints on real-time implementation in terms of what (and how much) information needs to be communicated as well as how often it needs to be communicated. In turn, these requirements dictate what type of communication infrastructure you need to communicate between the base and rover stations.
- It is therefore common to adopt a different approach. Specifically, at the base station, you can compute the geometric range term directly since both the receiver and satellite positions are known. As such, we can generate a correction term as follows (subscript 'b' is for the base station)

$$\begin{aligned} P_b - \rho_b &= d\rho_b + c(dT - dt_b) + d_{iono,b} + d_{trop,b} + m_{P,b} + n_{P,b} \\ &= \text{Correction} \end{aligned}$$

Alternate Between-Receiver Difference (2/2)

- Note that the correction term only contains error terms. Since most of the errors change slowly, the corrections can be generated (and communicated) at a relatively low rate. This eases the requirements of the communication system.
- Once the corrections are received at the rover station, they can be applied to the rover data (subscript ‘r’ is for rover station)

$$\begin{aligned}
P_r &= \rho_r + d\rho_r + c(dT - dt_r) + d_{iono,r} + d_{trop,r} + m_{P,r} + n_{P,r} - \text{Correction} \\
&= \rho_r + d\rho_r + c(dT - dt_r) + d_{iono,r} + d_{trop,r} + m_{P,r} + n_{P,r} \\
&\quad - d\rho_b + c(dT - dt_b) + d_{iono,b} + d_{trop,b} + m_{P,b} + n_{P,b} \\
&= \rho_r + \Delta d\rho + c(\Delta dT - \Delta dt) + \Delta d_{iono} + \Delta d_{trop} + \Delta m_P + \Delta n_P
\end{aligned}$$

- In this case, the errors are the same as if the measurements were differenced directly.

Between-Satellite Single Differencing (1/2)

- The between-satellite single difference measurement equations can be written as

$$\begin{aligned}\nabla P^{AB} &= P^{B} - P^{A} \\ &= \nabla\rho + \nabla d\rho + c(\nabla dT - \nabla dt) + \nabla d_{iono} + \nabla d_{trop} + \nabla m_{P} + \nabla n_{P} \\ \nabla\Phi^{AB} &= \Phi^{B} - \Phi^{A} \\ &= \nabla\rho + \nabla d\rho + c(\nabla dT - \nabla dt) - \nabla d_{iono} + \nabla d_{trop} + \nabla m_{\Phi} + \nabla n_{\Phi} + \lambda\nabla N\end{aligned}$$

- What can we say about the errors?

Error	Magnitude
$\nabla d\rho$	
$c\nabla dT$	
$c\nabla dt$	
∇d_{iono}	
∇d_{trop}	

Error	Magnitude
∇m_{P}	
∇n_{P}	
∇m_{Φ}	
∇n_{Φ}	

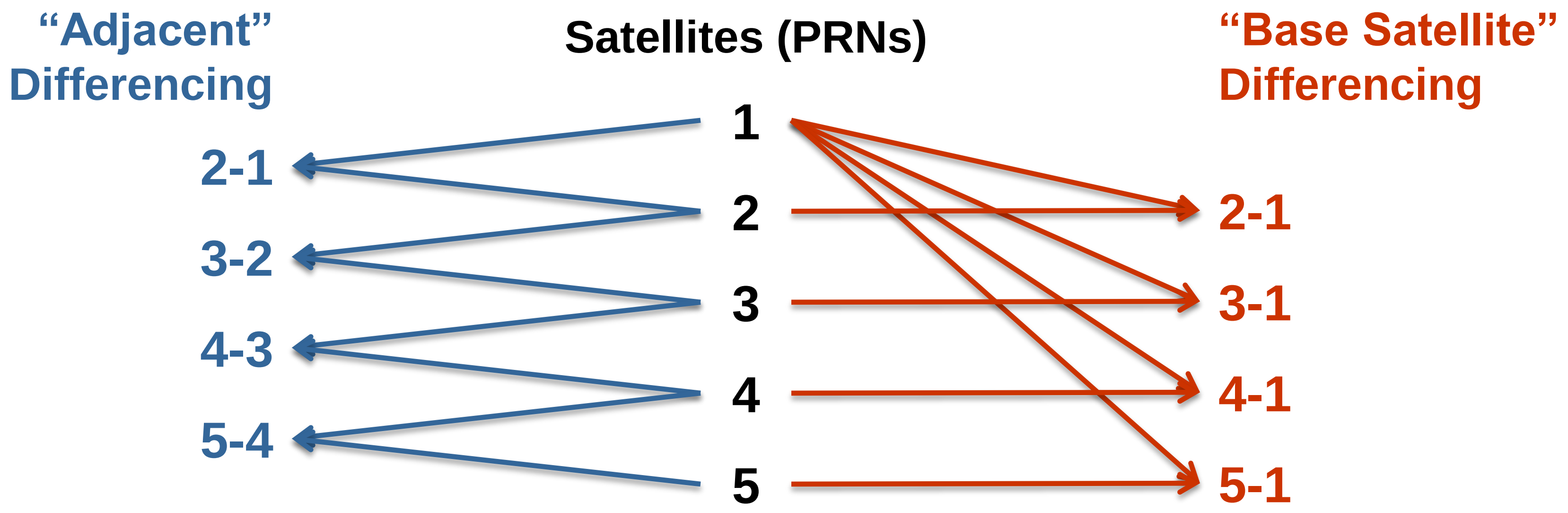
Between-Satellite Single Differencing (2/2)

- Given what we know about the between-satellite errors, what are the unknowns in this case?

- Based on this, would you expect better positioning accuracy? Why or why not?

Measurement Correlation (1/2)

- Assuming you have M un-differenced measurements at a given epoch, you can form a M-1 *independent* between-satellite single differences.
  - In theory, you can difference between any two satellites but it is common to select one satellite as the "base satellite" and difference all observations relative to the base satellite.



Measurement Correlation (2/2)

- Regardless of how you form the between-satellite difference, you introduce correlation between the measurements. Using the base satellite approach:

$$\begin{bmatrix} \Delta\ell_{12} \\ \Delta\ell_{13} \\ \Delta\ell_{14} \\ \Delta\ell_{15} \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \\ \ell_5 \end{bmatrix} \quad \Rightarrow \quad C_{D\ell} = B \cdot C_\ell \cdot B^T$$

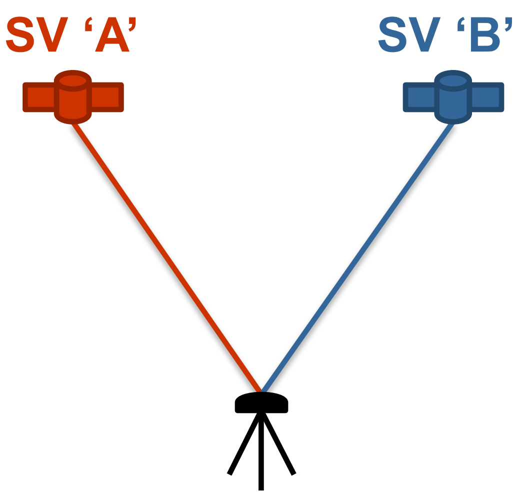
$$\Delta\vec{\ell} = B \cdot \vec{\ell}$$

where ℓ can be pseudorange or carrier phase. Then, we can write

$$C_\ell = s_\ell^2 \cdot I \qquad \Rightarrow \quad C_{D\ell} = \begin{cases} 2s_\ell^2 & \text{diagonal} \\ s_\ell^2 & \text{off-diagonal} \end{cases}$$

Summary of Between-Satellite Differencing

- The primary benefit of between-satellite differencing is that is removes the receiver clock offset, although this does not help to improve solution accuracy.
- As with between-receiver differencing, the uncorrelated errors (noise and multipath) are increased by $\sqrt{2}$.
- Between-satellite differences are correlated with each other.
- The primary use of between-satellite single differences is to help form double differences (discussed shortly).



Double Differencing

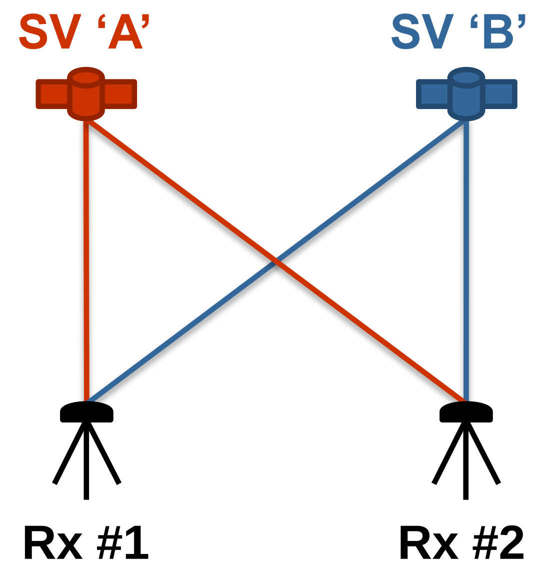
Double Differencing

- Double differencing combines between-receiver and between-satellite differences. The measurement equations are (simplifications made based on discussions in previous slides)

$$\begin{aligned}\nabla\Delta P_{12}^{AB} &= \Delta P_{12}^{B} - \Delta P_{12}^{A} = \nabla P_{2}^{AB} - \nabla P_{1}^{AB} \\ &= \nabla\Delta\rho + \nabla\Delta d\rho + \nabla\Delta d_{iono} + \nabla\Delta d_{trop} + \nabla\Delta m_{P} + \nabla\Delta n_{P}\end{aligned}$$

$$\begin{aligned}\nabla\Delta\Phi_{12}^{AB} &= \Delta\Phi_{12}^{B} - \Delta\Phi_{12}^{A} = \nabla\Phi_{2}^{AB} - \nabla\Phi_{1}^{AB} \\ &= \nabla\Delta\rho + \nabla\Delta d\rho + \nabla\Delta d_{iono} + \nabla\Delta d_{trop} + \nabla\Delta m_{\Phi} + \nabla\Delta n_{\Phi} + \lambda\nabla\Delta N_{12}\end{aligned}$$

- Combines the aspects of both single differences:
  - Reduces spatially correlated errors
  - Removes receiver and satellite clocks
  - Increases the standard deviation of multipath and noise by 2



Double Difference Pseudorange

- Since the receiver clock has been removed and the base station receiver (i.e., receiver #1) is assumed to have known coordinates, the only unknowns are the position of the rover receiver (x,y,z or ϕ,λ,h).
- Expanding the double difference geometric range term gives

$$\begin{aligned}\Delta\nabla\rho &= \left(\rho_2^B - \rho_1^B\right) - \left(\rho_2^A - \rho_1^A\right) \\ &= \left|\vec{r}_{SV}^{\,B} - \vec{r}_{Rx2}\right| - \left|\vec{r}_{SV}^{\,B} - \vec{r}_{Rx1}\right| - \left|\vec{r}_{SV}^{\,A} - \vec{r}_{Rx2}\right| + \left|\vec{r}_{SV}^{\,A} - \vec{r}_{Rx1}\right|\end{aligned}$$

- The design matrix is thus given by

$$\begin{aligned}A_{\nabla\Delta P,i} &= \left.\begin{bmatrix} \dfrac{\partial\nabla\Delta P^{AB}}{x_{Rx2}} & \dfrac{\partial\nabla\Delta P^{AB}}{y_{Rx2}} & \dfrac{\partial\nabla\Delta P^{AB}}{z_{Rx2}} \end{bmatrix}\right|_{\mathbf{x}=\hat{\mathbf{x}}} \\ &= \begin{bmatrix} \left(\dfrac{\left(\hat{x}_{Rx2} - x_i^B\right)}{\hat{\rho}_i^B} - \dfrac{\left(\hat{x}_{Rx2} - x_i^A\right)}{\hat{\rho}_i^A}\right) & \left(\dfrac{\left(\hat{y}_{Rx2} - y_i^B\right)}{\hat{\rho}_i^B} - \dfrac{\left(\hat{y}_{Rx2} - y_i^A\right)}{\hat{\rho}_i^A}\right) & \left(\dfrac{\left(\hat{z}_{Rx2} - z_i^B\right)}{\hat{\rho}_i^B} - \dfrac{\left(\hat{z}_{Rx2} - z_i^B\right)}{\hat{\rho}_i^A}\right) \end{bmatrix}\end{aligned}$$

Double Difference Carrier Phase

- For the carrier phase, the unknowns are the position and the double difference ambiguities. The design matrix is as follows (in hyper-matrix notation)

$$A_{\nabla\Delta\Phi,i} = \left[\begin{array}{c:c} A_{\nabla\Delta P,i} & \underbrace{0 \;\; \cdots \;\; \lambda \;\; \cdots \;\; 0}_{\text{For ambiguity states}} \end{array}\right]_{\mathbf{x}=\hat{\mathbf{x}}}$$

For position states
(same as for DD pseudorange)

- In this case, the matrix is better conditioned than the between-receiver single difference case. In fact, this is the key reason for using the double difference in the first place. Specifically, it allows the receiver clock and the ambiguity states to be separated (by eliminating the former altogether).

Unless otherwise stated, we henceforth assume we are using double differencing when using carrier phase data

Measurement Correlation

- As with the between-satellite single difference, the formation of double differenced observations introduces a correlation between measurements.

- Again, you can difference between any two satellites but it is common to select one satellite as the "base satellite" and difference all observations relative to the base satellite. In this case, assuming equal (un-differenced) measurement variances, the covariance matrix of the double difference measurements is

$$C_\ell = s_\ell^2 \cdot I \qquad \Rightarrow \quad C_{\nabla\Delta\ell} = \begin{cases} 4s_\ell^2 & \text{diagonal} \\ 2s_\ell^2 & \text{off-diagonal} \end{cases}$$

Static Double Difference Cycle Slip Detection (1/2)

- Difference between the measured double difference carrier phase and the computed double difference range can be formed as follows at epoch k

$$w_k = \nabla\Delta\Phi_k - \nabla\Delta\rho_k \approx \lambda\nabla\Delta N$$

- Since N is constant over time (assuming no cycle slips), at epoch k+1

$$w_{k+1} = \nabla\Delta\Phi_{k+1} - \nabla\Delta\rho_{k+1} \approx \lambda\nabla\Delta N$$

- Any difference between w_k and w_{k+1} would indicate the presence of cycle slips. For example, if $(w_k\text{-}w_{k+1})$ = 0.191 m, this would indicate that there has been a slip of 1 cycle on L1. However, it does not indicate on which satellite (base or non-base) or which receiver (base or rover) but it is not usually important to isolate this specific information.

Static Double Difference Cycle Slip Detection (2/2)

- It is noted that this method assumes that $\Delta\nabla\rho$ is known with a relatively good accuracy which implies that the remote's position has been determined in the first place!

- This issue can be resolved by first performing a triple difference solution to generate a good approximation of the remote's position (discussed next).
  - The triple difference solution is then used to compute $\Delta\nabla\rho$
  - Used in special commercially available software packages
  - Only feasible for static mode

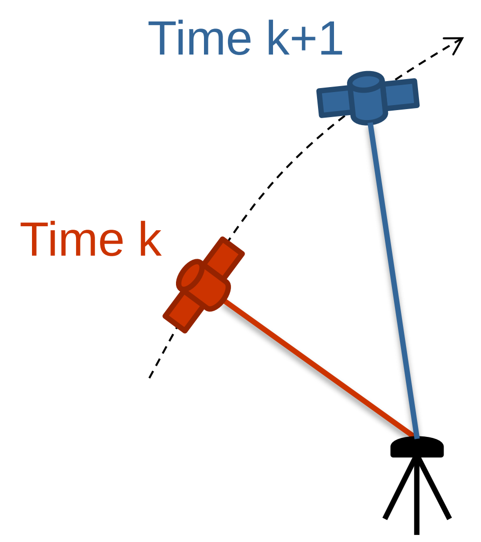
Time Differencing

Time Differences

- In some instances, differences over time can be used. They are typically used with the carrier phase, and the corresponding measurement equation is as follows

$$\delta\Phi(t_1,t_2) = \Phi(t_2) - \Phi(t_1)$$
$$= \delta\rho + \delta d\rho + c(\delta dT - \delta dt) - \delta d_{iono} + \delta d_{trop} + \delta m_{\Phi} + \delta n_{\Phi} + \cancel{\lambda\delta N}$$

- If there are no cycle slips, the ambiguity term disappears. This is the main advantage of this approach.
- The noise and multipath are increased by $\sqrt{2}$ relative to the un-differenced case. All other error terms reduce to the *change* in error over time (usually small).
- This difference can also be used to compute *changes* in position over time.



Triple Differences

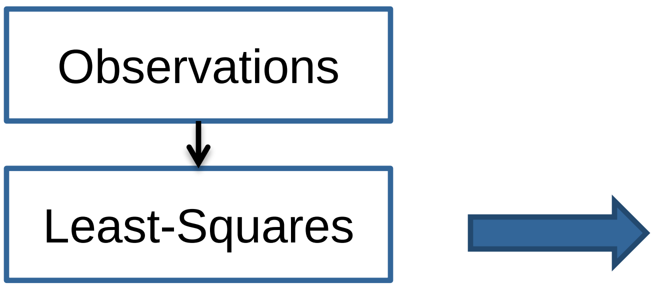
- A triple difference is formed by differencing two double differenced measurements across epochs. As should be expected:
  - All clock errors and the ambiguities are eliminated (assuming no cycle slips).
  - Spatially correlated errors are reduced.
  - The standard deviation of multipath and noise and increased by $2\sqrt{2}$.

- In some highly specialized static processing software packages, this approach is used as an initial processing step. The downside is that you require a fairly large amount of data (few hours) to get a reasonably accurate position solution.
  - To see why this is, derive the design matrix and show that for short time intervals, the elements are nearly zero.

Ambiguity Resolution: Concept

Why Ambiguity Resolution? (1/3)

- The carrier phase ambiguities are known to be integers. However, estimating them in a least-squares adjustment yields real-valued estimates – often called ***float ambiguities***.



These are real-valued estimates
(least-squares does not know any better)

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{x} & \hat{y} & \hat{z} & \nabla\Delta\hat{N}_{12} & \cdots & \nabla\Delta\hat{N}_{1M} \end{bmatrix}^T$$

- Estimating float ambiguities reduces the degrees of freedom in your adjustment resulting in poorer performance.
  - Although the real-valued ambiguity estimates will (theoretically) converge to the true integer values, this can take a long time (hours) and the process has to reset if there is a cycle slip or loss of lock.

Why Ambiguity Resolution? (2/3)

- ***Ambiguity resolution*** exploits the integer nature of the ambiguities. Specifically, once resolved as integers – that is, once the (hopefully correct) integer values are determined – the ambiguities no longer need to be estimated and they can be removed from the vector of unknowns, resulting in more degrees of freedom and higher accuracy.
- If the ambiguities are known, we can write:

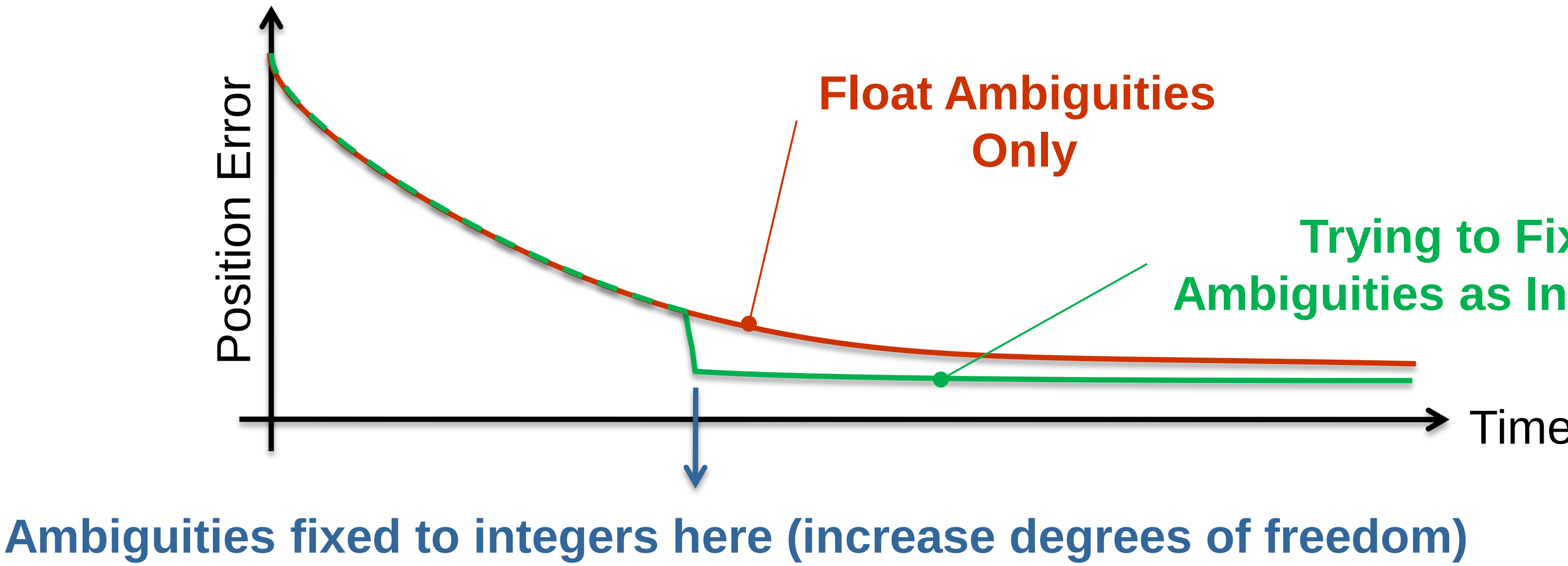
$$\nabla\Delta\Phi - \lambda\nabla\Delta N = \underbrace{\nabla\Delta\rho + \nabla\Delta d\rho + \nabla\Delta d_{iono} + \nabla\Delta d_{trop} + \nabla\Delta m_{\Phi} + \nabla\Delta n_{\Phi}}$$

This is the same as the right hand side of the double difference pseudorange equation, but with smaller noise and multipath errors! (and opposite sign on the ionosphere)

- In this way, the best possible positioning accuracy is obtained more quickly than when using float ambiguities.

Why Ambiguity Resolution? (3/3)

- Once the ambiguities are resolved, we say they are "fixed" (to integer values) and the position accuracy improves ***if the ambiguities are resolved correctly***.
  - A wrong ambiguity fix is the same as a ranging error whose magnitude is an integer multiple of carrier phase wavelengths, and will consequently produce positioning errors.
  - Wrong fixes are difficult to detect and pose a serious integrity threat because the estimated accuracy will be less than the actual error.



Getting a Feel for Ambiguity Resolution (1/2)

- Ambiguity resolution is a very challenging process for two main reasons: (i) it is computationally difficult, and (ii) there is a possibility of incorrectly fixing an ambiguity. The latter comes with an associated integrity risk. Incorrect ambiguity fixing is a major concern because of the short wavelengths involved.

Rule of Thumb:
To resolve carrier phase ambiguities, you need your range errors (biases + noise) to be less than half of the carrier wavelength

- For L1 ($\lambda \approx 19$ cm), the above rule of thumb requires you to have range errors less than 9.5 cm!
- Note that the above “threshold” can be satisfied over time by trying to average out the various error sources.

Getting a Feel for Ambiguity Resolution (2/2)

- Given the rule of thumb on the previous page, we re-examine the double difference carrier phase equation

$$\nabla\Delta\Phi = \underbrace{\nabla\Delta\rho + \lambda\nabla\Delta N}_{\text{Desired parameters}} + \underbrace{\nabla\Delta d\rho + \nabla\Delta d_{iono} + \nabla\Delta d_{trop} + \nabla\Delta m_{\Phi} + \nabla\Delta n_{\Phi}}_{\text{Errors}}$$

- The importance of understanding the differential errors is now more important than ever. Without this knowledge, the following questions could not be answered:
  - What is the longest baseline over which I can resolve my ambiguities?
  - How large can the ionosphere error be before ambiguity resolution is unfeasible?
  - How long do I have to occupy a point in order to resolve my ambiguities?

Key Factors Affecting Ambiguity Resolution

- Static versus kinematic
  - Ambiguities are generally easier to resolve in static mode because the errors can be more effectively averaged
- Baseline separation
  - The shorter the baseline, the easier it is to resolve ambiguities
- Multipath
  - Since multipath is site-specific it has a major impact on ambiguity resolution
- Length of data set and geometry
  - Information is gained through the satellite geometry change – longer observation times give better opportunity for resolution
  - The more satellites tracked the better
- Type of GNSS receiver
  - Dual-frequency receivers can usually resolve ambiguities faster than single frequency systems due to ability to form linear carrier phase combinations

Ambiguity Resolution: *Some* Details

Static vs. Kinematic Processing

- Although the measurement equations are the same for static and kinematic positioning applications, there is a dramatic difference in terms of ambiguity resolution performance.

- To illustrate, consider tracking 10 satellites for one minute with a data rate of one measurement per second:

# Observations	# Unknowns (Static)	# Unknowns (Kinematic)
(10-1) DD / epoch x 60 epochs = 540 observations	3 position states + (10-1) DD ambiguities = 12 unknowns	3 position states / epoch x 60 epochs + (10-1) DD ambiguities = 189 unknowns
Degrees of freedom →	**528**	**351**

Static Processing

- For a static user, the solution can be obtained using a batch least-squares approach (sequential least-squares could also be used):

$$A = \begin{bmatrix} A(t_1) \\ A(t_2) \\ \vdots \\ A(t_{10}) \end{bmatrix}$$

Design matrix from epoch t_1

Design matrix from epoch t_2

Design matrix from epoch t_{10}

where, for M satellites (without cycle slips):

$$A(t_1) = \left[\left(\frac{\partial \nabla\Delta\vec{\Phi}(t_1)}{\partial \vec{r}} \right)_{(M-1)x3} \;\vdots\; \lambda I_{(M-1)x(M-1)} \right]$$

- Effectively, you average the observations together to get the best estimate of the unknowns.

Kinematic Positioning (1/2)

- For a kinematic user, the state vector contains a 3-vector of positions for *every epoch* plus the ambiguity states (one per DD ambiguity). As such, the least-squares design matrix is as follows

$$A = \begin{array}{cl}
\begin{bmatrix}
\left(\dfrac{\partial \nabla\Delta\vec{\Phi}(t_1)}{\partial \vec{r}_1}\right)_{(M-1)\times 3} & 0 & 0 & & \lambda I_{(M-1)\times(M-1)} \\
0 & \left(\dfrac{\partial \nabla\Delta\vec{\Phi}(t_2)}{\partial \vec{r}_2}\right)_{(M-1)\times 3} & 0 & & \lambda I_{(M-1)\times(M-1)} \\
0 & 0 & \left(\dfrac{\partial \nabla\Delta\vec{\Phi}(t_3)}{\partial \vec{r}_3}\right)_{(M-1)\times 3} & & \lambda I_{(M-1)\times(M-1)} \\
0 & 0 & 0 & \ddots & \vdots
\end{bmatrix}
&
\begin{matrix} \text{Epoch 1} \\ \\ \text{Epoch 2} \\ \\ \text{Epoch 3} \\ \vdots \end{matrix}
\\
\begin{matrix} \text{Epoch 1 Position} & \text{Epoch 2 Position} & \text{Epoch 3 Position} & \text{Ambiguities (same for all epochs assuming no cycle slips)} \end{matrix} &
\end{array}$$

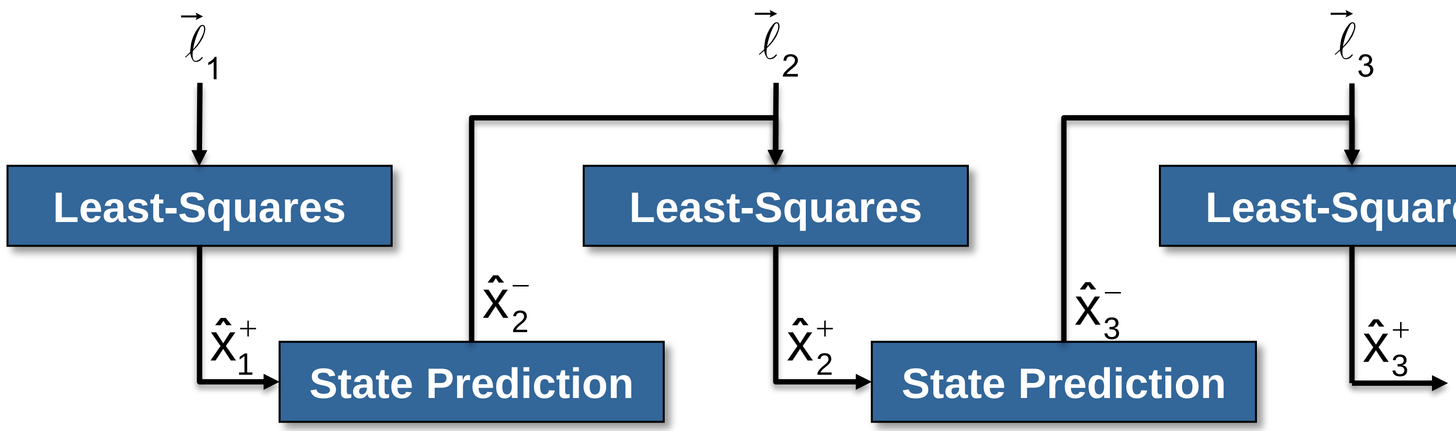
Kinematic Positioning (2/2)

- From the design matrix on the previous slide, every epoch's position estimate is largely independent of every other epoch's position estimate. In reality, this is not usually the case. That is, we can usually put some constraint on where a user is located at time t_{k+1} given their position at time t_k. As a result, using a "simple" least-squares approach will yield suboptimal results.
- To get around this, kinematic positioning algorithms use a ***Kalman filter***. The concept of a Kalman filter is presented on the following slides in order to give students a feel for what is happening.
  - Details of Kalman filtering are left for ENGO 563 and/or ENGO 585.
  - A high-level discussion of the differences/similarities between least-squares and Kalman filtering is

    Petovello, M.G. (2013) "*What are the differences between least-squares and Kalman filtering?*", Inside GNSS, Volume 8, Number 2, pp. 20-22. (Available at: http://insidegnss.com/solutions – see the Mar/Apr 2013 issue)

Brief Introduction to the Kalman Filter (1/3)

- Consider the situation where the position of a user can be predicted based on a previous position solution (possibly combined with some velocity information). The estimation process can then be illustrated as follows:



- The “state prediction” is based on some known/assumed knowledge of how the position changes with time. Once a prediction is available, it can be used to weight the states in a least-squares solution.

Brief Introduction to the Kalman Filter (2/3)

- After the first epoch, the least-squares solution can thus be written as

$$\Delta\vec{x}_{k+1} = \left(A_{k+1}^{T} C_{\ell}^{-1} A_{k+1} + C_{x^-}^{-1}\right)^{-1} A_{k+1}^{T} C_{\ell}^{-1} w_{k+1}$$

where the C_x term is the covariance matrix of the *predicted* states. This can be approximately expressed as

$$C_{x^-,k+1} = C_{x^+,k} + DC$$

From previous epoch's least-squares solution

Uncertainty in the state prediction

- The magnitude of ΔC dramatically affects the overall solution.

Brief Introduction to the Kalman Filter (3/3)

- For a static user the state prediction is perfect, in which case $\Delta C = 0$. In this case, it can be shown that the result is the same as the batch least-squares solution.
- However, if the user's position is highly <u>un</u>predictable ΔC will tend to infinity and the result is a least-squares solution:

$$\Delta\vec{x}_{k+1} = \left(A_{k+1}^{T} C_{\ell}^{-1} A_{k+1} + \cancel{C_{x^{-}}^{-1}} \right)^{-1} A_{k+1}^{T} C_{\ell}^{-1} w_{k+1} = \left(A_{k+1}^{T} C_{\ell}^{-1} A_{k+1} \right)^{-1} A_{k+1}^{T} C_{\ell}^{-1} w_{k+1}$$

- Although the above equations are written in terms of least-squares, a Kalman filter accomplishes the same task. In other words, a Kalman filter is a fancy form of least-squares.
  - **Note**: In addition to computing a least-squares solution, the formal Kalman filter equations also provide a means of parameterizing the state prediction and quantifying ΔC. Students taking ENGO 563 and/or ENGO 585 will see more details on Kalman filtering.

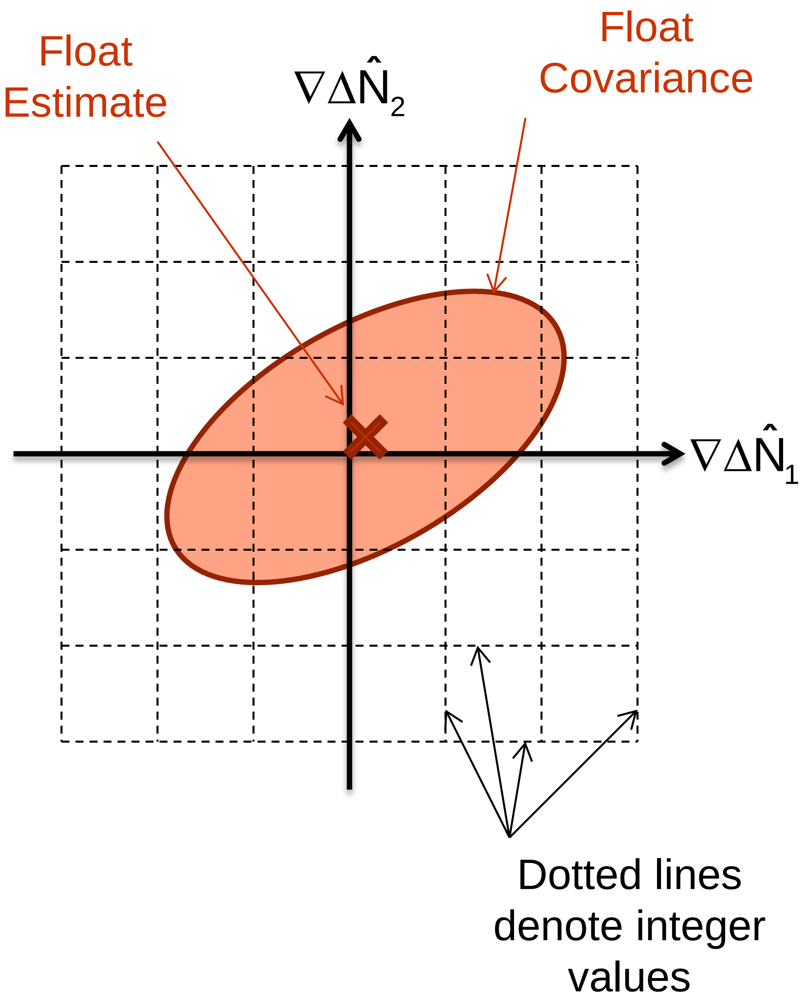
Steps to Ambiguity Resolution

- Although a variety of ambiguity resolution algorithms have been developed, they all share the same basic steps, as listed below and summarized on the following slides:
    1. Estimate the position and the float ambiguities using least-squares (static) or a Kalman filter (kinematic)
    2. Define a search space within which the integer ambiguities are *assumed* to reside
    3. For each ambiguity combination, compute the least-squares residuals
    4. Use the residuals to identify the best ambiguity set and determine if they are reliable; if so, use this set as the ***fixed*** integer ambiguities and compute a fixed ambiguity position solution

- It is also noted that if a cycle slip is detected, the ambiguity resolution process for the affected ambiguity(ies) must be restarted (i.e., you need to estimate a new ambiguity).

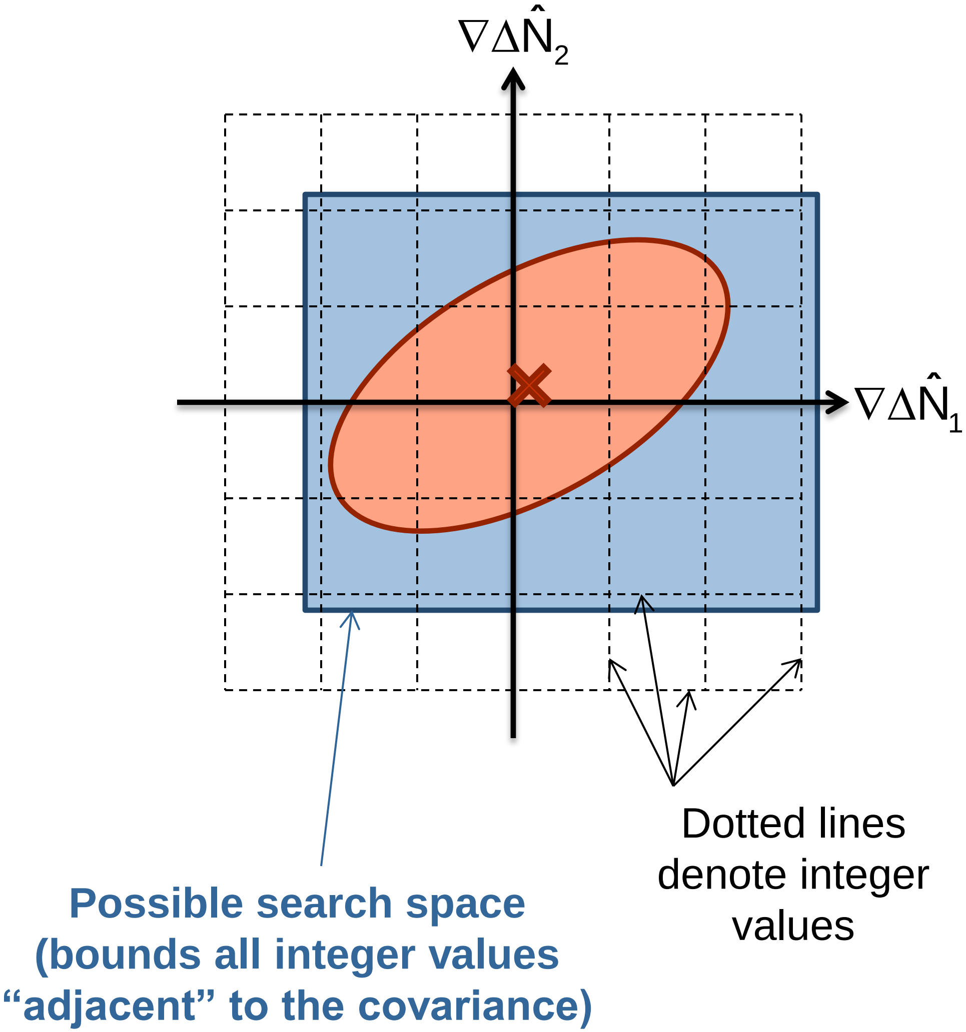
Step 1: Estimate the Float Ambiguities

- This was discussed on earlier slides but to recap, the ambiguities are estimated as unknown parameters using a least-square adjustment or a Kalman filter. The output of this step includes:
  - Estimates of the float ambiguities for all satellites
  - Covariance matrix for the estimated ambiguities
  - Position estimate based on float ambiguities ("float solution")



Step 2: Define an Ambiguity Search Space (1/2)

- The second step is to define a ***search space*** that (hopefully) contains the true integer ambiguities.
  - Assuming there are M ambiguities being estimated, the search space will be M-dimensional. However, to illustrate, consider a simplified case (at right) where only two ambiguities are being estimated.
- As can be seen, the covariance matrix of the float ambiguities is used to define the search space.

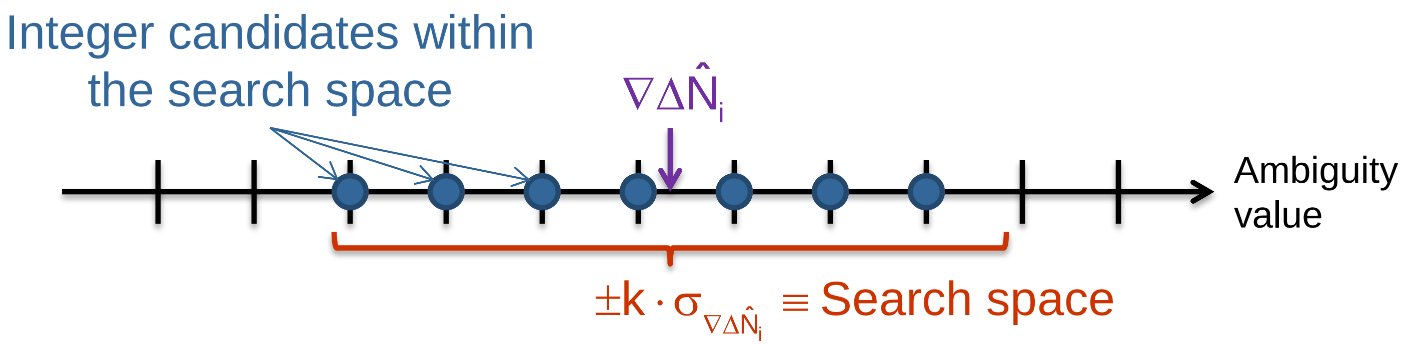


Step 2: Define an Ambiguity Search Space (2/2)

- Although the plot on the previous slide illustrates the basic concept, mathematically, the search space (for each ambiguity) is computed as

$$\nabla D\hat{N} - k \cdot s_{\nabla D\hat{N}} \leq \nabla DN_{INT} \leq \nabla D\hat{N} + k \cdot s_{\nabla D\hat{N}}$$

where k is a scalar such that the search range is ±kσ wide (e.g., if k=3, the search space will contain the true ambiguity with a probability of 99.7%).



- The challenge with this is that the search space can become very large. For example, if each ambiguity has 10 possible values and there were seven ambiguities, there are a total of 10^7 possible combinations!
  - A lot of research has gone into reducing the search space. These techniques are not discussed in this course, but they offer tremendous computational savings and are used by all major GNSS processing software.

Step 3: Compute Residuals

- For every integer ambiguity combination in the search space, use the candidate ambiguities to compute an updated position. Also compute residuals for each ambiguity combination.
  - Recall that the residuals quantify how well the observations “fit together”, with smaller residuals *suggesting* a better fit. They can be computed as:

$$\vec{v} = A\vec{\delta} + \vec{w}$$

Geometric range computed using float position estimate

$$\vec{w} = \vec{\ell} - f\left(\hat{\vec{x}}\right)$$

$$= \left(\vec{\Phi} - \lambda \nabla\Delta \hat{\vec{N}}_{INT}\right) - \nabla\Delta\hat{\vec{\rho}}$$

Phase measurements

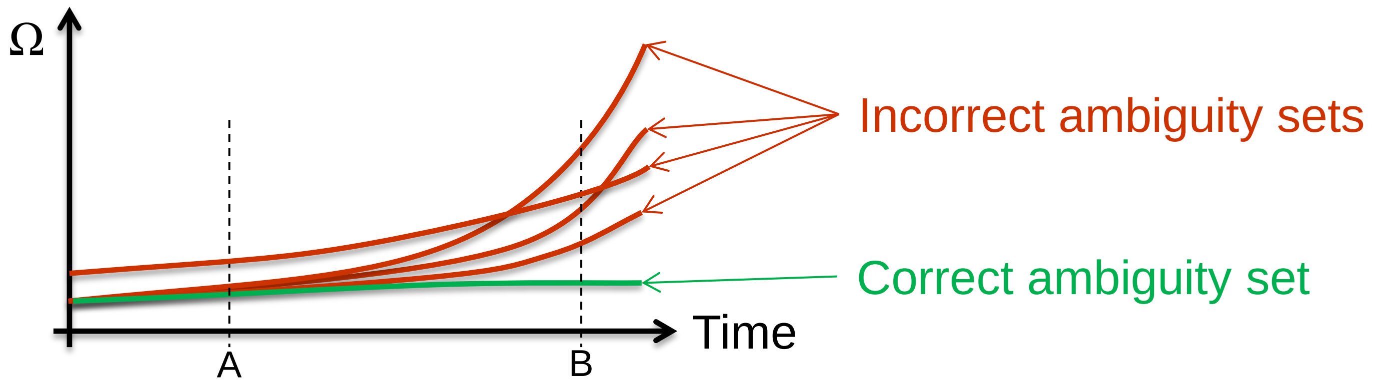
Integer ambiguity candidate

- Since we are interested in all residuals (as a whole), we compute the sum of squared (SOS) residuals (Ω):

$$\Omega = \vec{v}^T C_\ell^{-1} \vec{v}$$

Step 4: Select the Integer Ambiguity Set (1/2)

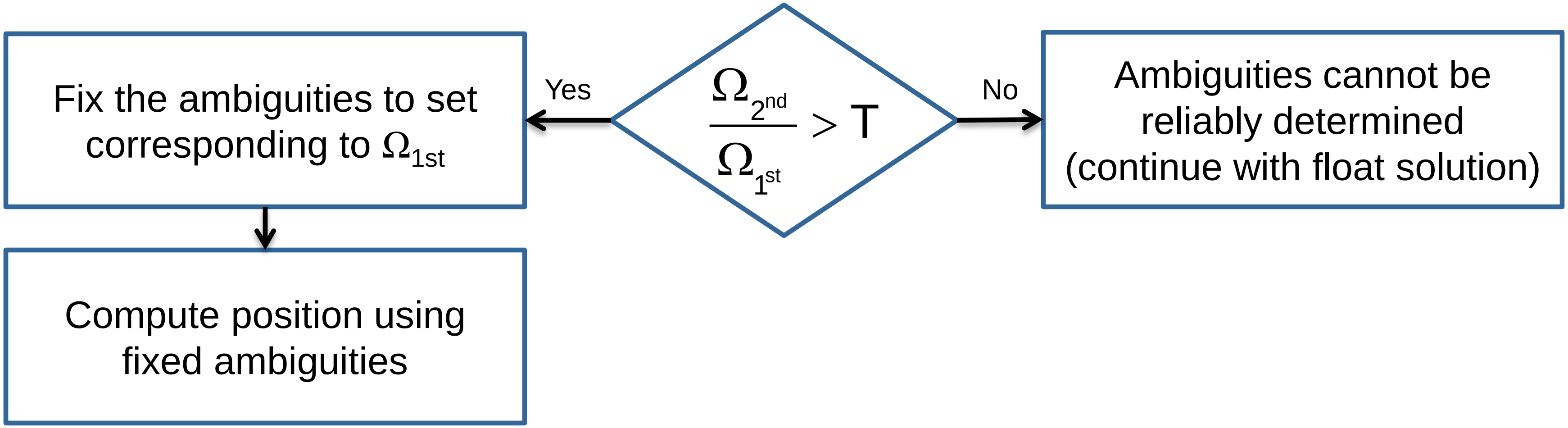
- The SOS values from step 3 will increase with time for incorrect ambiguity combinations. However, they will tend to remain constant for the correct ambiguity set.



- To determine when to fix the ambiguities, the 2nd smallest SOS value is divided by the 1st smallest SOS value. When this ratio exceeds some threshold, the ambiguity set corresponding to the smallest SOS are assumed to be correct.

Step 4: Select the Integer Ambiguity Set (2/2)

- Schematically, the test is as follows



- A common value for the above threshold (T) is 2 or 3. However, some points to keep in mind:
  - Smaller thresholds will result in fixing the ambiguities faster (see point A on previous slide) but less reliably.
  - Larger thresholds require longer times to fix the ambiguities, but the result is more reliable (see point B on previous slide). By extension, the longer you wait, the more reliable your solution will be.

Ambiguity Validation

- Once the ambiguities are resolved as integers, the residuals must be continually monitored to make sure they remain sufficiently small. This is accomplished by performing the global test on the residuals (i.e., test on the variance) at every epoch.

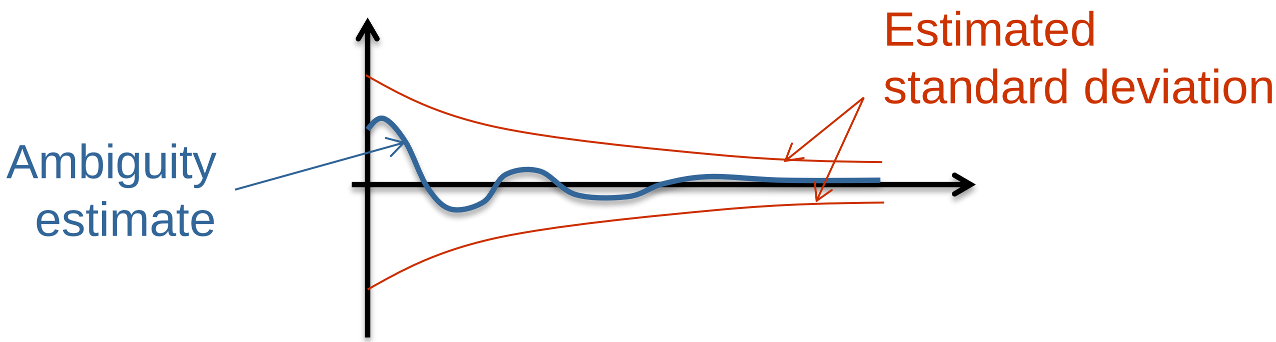
- If the residuals grow too large (usually over consecutive epochs to allow for short term multipath effects), it is likely that the ambiguity resolution process failed (i.e., the wrong ambiguity set was selected) and must be restarted again.

Maximizing Success of Ambiguity Resolution (1/3)

- Since an incorrect ambiguity fix (even by one cycle on one satellite) will cause the position error to far exceed the estimated position accuracy (i.e., obtained from C_x), it follows that success of the ambiguity resolution process should be maximized.
- The bottom line is that the longer you wait to resolve the ambiguities, the greater the chance of success. This happens for two reasons
  - Float ambiguities become decorrelated and tend to the true integer values
  - The larger the *change* in satellite geometry, the easier it is to identify incorrect ambiguity sets
- The above points are discussed in more detail on the following slides.

Maximizing Success of Ambiguity Resolution (2/3)

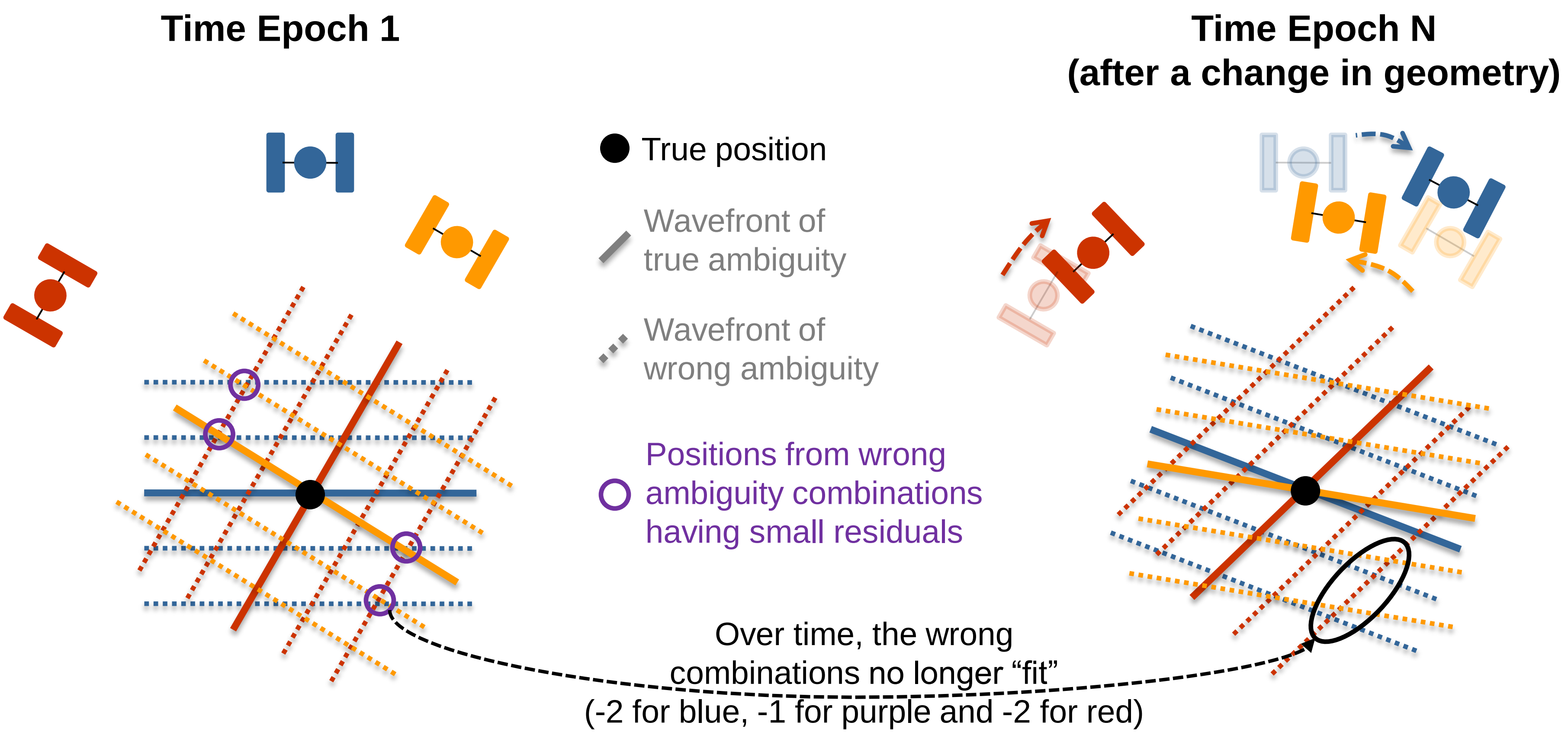
- The estimated float ambiguities are highly correlated with each other when limited data is used. Correspondingly, the estimated values tend to contain fairly large errors.



- Over time, as more data is used, the ambiguities decorrelate. In and of itself, this helps the ambiguity resolution process, but it also results in more accurate float ambiguity estimates.
  - **Note**: If enough time elapses (several hours), the float ambiguities will tend to the true fixed values. In this case, there is little (if any) need to resolve the ambiguities at all (i.e., use the float values instead).

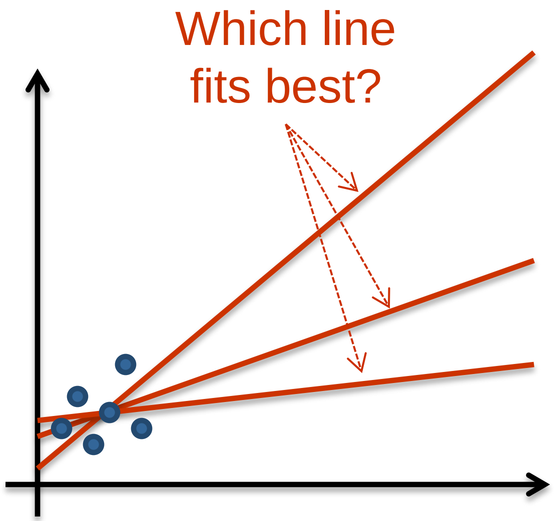
Maximizing Success of Ambiguity Resolution (3/3)

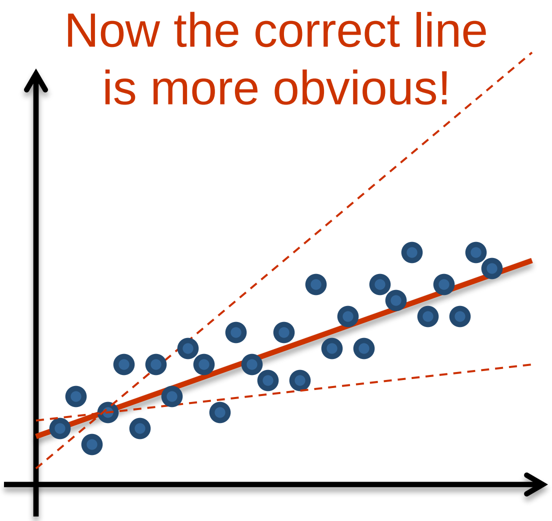
- The change in satellite geometry plays a very important role in ambiguity resolution. The more time elapses, the easier it becomes to identify incorrect ambiguity combinations.



Ambiguity Resolution Analogy

- Another way of thinking about the ambiguity resolution is to consider fitting a line to a set of points. If the points are closely spaced, it is difficult to determine the slope. However, if the points are spread apart the problem becomes easier.





- It should also be noted that the data rate is not very important (consider removing 2 of every 3 points above and whether that will make a big difference).

Sample Results

- Data was collected for 10 min on a 720 m baseline. A total of six satellites were tracked. The table below shows the results of the ratio test after processing 5 and 10 minutes.

Amount of Data	Smallest SOS	2nd Smallest SOS	SOS Ratio	Decision
5 min	0.137	0.155	1.13	Ratio is less than threshold (typically 3). Use float ambiguities.
10 min	0.044	0.386	8.8	Ratio exceeds threshold, so fix ambiguities to integers.

Expected Accuracy

- Assuming the ambiguities can be resolved as integers, what level of accuracy can be achieved?
  - The answer depends on the application, the level of errors, the satellite geometry and the amount of data collected.
  - Consider the measurement errors (per epoch) below:

Error Source	Typical Error (DD)
Orbit	0.05 ppm
Ionosphere	2 ppm
Troposphere	1 ppm
Multipath	(λ x 0.1) x 2 = 0.038 m (1σ)
Noise	1 mm x 2 = 0.002 m (1σ)

$$\sigma_{\nabla\Delta\Phi} = \sqrt{\sigma^2_{d\rho} + \sigma^2_{iono} + \sigma^2_{trop} + \sigma^2_{m} + \sigma^2_{n}}$$

$$= \sqrt{(2.24\ \text{ppm} \times b)^2 + (0.038\ \text{m})^2}$$

- For a 1 km baseline, assuming an HDOP of 1.2 and a VDOP of 1.8 (= 1.5 x HDOP), the expected positioning accuracy is **4.6 cm horizontally (DRMS)** and **6.9 cm vertically (1σ)**!

Linear Carrier Phase Combinations

Linear Phase Combinations

- Recall (from Chapter 5) that L1 and L2 carrier phase measurements can be linearly combined as

$$\phi_{a,b} = a\phi_{L1} + b\phi_{L2}$$

with a wavelength of

$$\lambda_{a,b} = \frac{c}{af_{L1} + bf_{L2}} = \frac{\lambda_{L1}\lambda_{L2}}{b\lambda_{L1} + a\lambda_{L2}}$$

- We now revisit this concept in more detail in the context of ambiguity resolution with consideration for
  - Errors in units of cycles (important for ambiguity resolution)
  - Errors in units of metres (important for positioning accuracy)

Reference: Petovello, M.G. (2009) **What are Linear Carrier Phase Combinations and What are the Relevant Considerations?**, Inside GNSS, Vol 4, No 1, pp. 16-19. Available at: http://insidegnss.com/node/1122

Some Common Combinations

Name	a	b	λ (m)	Ambiguity
L1 only	1	0	0.1903	N_{L1}
L2 only	0	1	0.2442	N_{L2}
Widelane	1	-1	0.8619	$N_{WL} = N_{L1} – N_{L2}$
Ionosphere-free	1	$-f_{L2} / f_{L1}$	0.4844	$N_{IF} = N_{L1} – f_{L2} / f_{L1} N_{L2}$
Narrowlane	1	1	0.1070	$N_{NL} = N_{L1} + N_{L2}$

Note that the ambiguities for the ionosphere free (IF) combination are no longer integer in nature and thus cannot be used in ambiguity resolution algorithms

Error Characterization

- For the purpose of this discussion, we can divide the errors into three general categories
  - ***Geometric errors*** are the same on all frequencies and include orbit and troposphere errors (clock errors can also be included, but these are removed during differencing)
  - ***Ionosphere errors*** which are treated separately because of their dispersive nature
  - ***Stochastic errors*** which include multipath and noise and are characterized by their standard deviation

Errors in Units of Cycles

Category	Error on L1	Error
Geometric	$\dfrac{G}{\lambda_{L1}}$	$\dfrac{G}{\lambda_{a,b}}$
Ionosphere	$I_{L1}^{[cyc]} = \dfrac{1}{\lambda_{L1}} \dfrac{40.3 \cdot TEC}{f_{L1}^2}$	$I_{a,b}^{[cyc]} = \left(\dfrac{af_{L2} + bf_{L1}}{f_{L2}} \right) \cdot I_{L1}^{[cyc]}$
Stochastic	$\sigma^{[cyc]}$	$\sigma_{a,b}^{[cyc]} = \sqrt{a^2 + b^2} \cdot \sigma^{[cyc]}$

Notes

- Choosing a & b to give longer wavelengths, makes the geometric error smaller
- The stochastic errors are assumed to be units of cycles; noise is a few percent of a cycle and multipath is known to be wavelength-dependent

Errors in Units of Length

Multiplying the results from the previous slide by the corresponding wavelength gives the errors in units of length (e.g., metres)

Category	Error on L1	Error
Geometric	G	G
Ionosphere	$I_{L1} = \dfrac{40.3 \cdot TEC}{f_{L1}^2}$	$I_{a,b} = \left(\dfrac{af_{L2} + bf_{L1}}{af_{L1} + bf_{L2}} \cdot \dfrac{f_{L1}}{f_{L2}} \right) \cdot I_{L1}$
Stochastic	$\lambda_{L1} \sigma^{[cyc]}$	$\sigma_{a,b} = \lambda_{a,b} \cdot \sqrt{a^2 + b^2} \cdot \sigma^{[cyc]}$

Notes

- Geometric errors are unaffected by the selection of a & b
- The stochastic errors are assumed to be functions of the wavelength; noise is a few percent of a cycle and multipath is known to be wavelength-dependent

Selecting a Linear Combination

- Analysis of the previous slides will show that there is a tradeoff when forming linear combinations
  - It is generally not possible to reduce errors both in units of cycles and units of length
  - Some combinations will allow for easier and more robust ambiguity resolution, but will likely give poorer position accuracy (larger errors in units of length) and vice versa

- Selecting a linear combination is therefore not straightforward and the decision is often application specific

Estimation Options for Carrier Phase Processing

Strategy	Ambiguity	Observables	Ionosphere Bias
1: Single Frequency	N_1	CP_1, P	Not Parameterized
2: Wide Lane (WL)	N_{WL}	CP_1, CP_2, P	
3: L1 and L2	$N_1 \; N_2$	CP_1, CP_2, P	
4: L1 and WL	$N_1 \; N_{WL}$	CP_1, CP_2, P	
5: IF fixed	$N_1 \; N_{WL}$ (IF Fixed)	CP_1, CP_2, P	Ionosphere-Free Combination
6: IF Float	N_{IF} (IF Float)	CP_1, CP_2, P	
7: L1 and L2 with SIM	$N_1 \; N_2 \; I_1$	CP_1, CP_2, P	Stochastic Ionosphere Modeling
8: L1 and WL with SIM	$N_1 \; N_{WL} \; I_1$	CP_1, CP_2, P	

CP_1 – L1 Carrier Phase
CP_2 – L2 Carrier Phase
P – Pseudorange Code
SIM - Stochastic Ionosphere Modeling

LAMBDA decorrelation and search is used to fix ambiguities when applicable

Reference: Liu, J. (2003), **Implementation and Analysis of GPS Ambiguity Resolution Strategies in Single and Multiple Reference Scenario**, M.Sc. thesis, UCGE Report No. 20168, Geomatics Engineering, University of Calgary.

The above is an example of the different options available for processing. The final decision will be based on the level of errors, the amount of data, the type of data (L1 vs. dual frequency), the baseline length, etc.

Precise Point Positioning (PPP)

PPP Introduction

- Precise Point Positioning (PPP) – not to be confused with the Precise Positioning Service (PPS) – is a means of obtaining high-accuracy position estimates <u>without</u> a base station

- Key attributes of PPP are as follows:
  - Absolute positioning with high accuracy
  - Absence of differencing means that all error sources need to be compensated in other ways
  - No need for another reference receiver and communication link
  - Precise satellite orbits and clock corrections are needed
  - Need advanced GPS error calibration methods
  - Decimetre and up to centimetre accuracy
  - Long convergence time is the primary challenge

Updated Observation Equations

- For PPP, the pseudorange and carrier phase observations are updated as follows

$$P = \rho + d\rho + c(dT - dt) + d_{ion} + d_{trop} + b_P^s - b_P^r + \varepsilon_P$$

$$\Phi = \rho + d\rho + c(dT - dt) - d_{ion} + d_{trop} + b_\Phi^s - b_\Phi^r + \varepsilon_\Phi + \lambda N$$

Satellite-based hardware biases

Receiver-based hardware biases

- Hardware biases were ignored until now because
  - They are typically much smaller than the other pseudorange errors
  - They are removed when forming double differences for carrier phase processing

Traditional PPP Model

- Traditionally, ionosphere-free code and phase combination is used in conjunction with precise orbit and clock information

$$P_{IF} = \frac{f_{L1}^2 P_{L1} - f_{L2}^2 P_{L2}}{f_{L1}^2 - f_{L2}^2}$$

$$= \rho - cdt + d_{trop} + b_{P_{IF}}^s - b_{P_{IF}}^r + \varepsilon_{P_{IF}}$$

$$\Phi_{IF} = \frac{f_{L1}^2 \Phi_{L1} - f_{L2}^2 \Phi_{L2}}{f_{L1}^2 - f_{L2}^2}$$

$$= \rho - cdt + d_{trop} + \frac{cf_{L1}N_{L1} - cf_{L2}N_{L2}}{f_{L1}^2 - f_{L2}^2} + b_{\Phi_{IF}}^s - b_{\Phi_{IF}}^r + \varepsilon_{\Phi_{IF}}$$

Atmospheric Error Mitigation

- Tropospheric error
  - Use Saastamoinen model to calculate zenith hydrostatic delay and zenith wet delay
  - Hard to precisely model the zenith wet delay that is thus usually estimated along with the position and clock terms; use a mapping function (e.g. Vienna mapping function) to project the zenith delay to the slant path

- Ionospheric error
  - Dual-frequency ionosphere-free combination to mitigate first-order error
  - Single-frequency user requires external ionosphere product (e.g. IONEX product by IGS)

Other Errors

- Satellite/receiver phase center offset and variation
  - Caused by difference between antenna mass and phase center
  - Corrections from, e.g. ANTEX product by IGS
  - Up to 2-3 metres offset for satellite, up to 20 cm for receiver
- Phase wind-up
  - Caused by satellite or receiver antenna rotation, up to a few centimetres
  - Critical for static point positioning to resolve integer ambiguity
  - Correction can be calculated by model
- Sagnac effect
  - Caused by earth rotation preventing the satellite and receiver clock synchronization
  - Up to hundreds of nanoseconds and can be corrected by model
- Tidal effect (solid Earth tides)
  - Corrected by models

Free Online PPP Services

- **CSRS-PPP**: Canadian Spatial Reference System, Natural Resources Canada
- **AUSPOS**: Geoscience Australia
- **GAPS**: University of New Brunswick
- **APPS**: Jet Propulsion Laboratory
- **SCOUT**: Scripps Orbit and Permanent Array Center (SOPAC), University of California, San Diego
- **magicGNSS**: GMV
- **CenterPoint RTX**: Trimble Navigation

Reference: GPS World, October 2013, http://gpsworld.com/a-comparison-of-free-gps-online-post-processing-services/

CSRS-PPP – Sample Results

- Data description
  - Location: CCIT Roof
  - Duration: 5 hours (Feb 2014)
  - Receiver: Trimble R10
  - Obs type: L1/P1/L2/P2

- Centimetre accuracy obtained after about 30 min