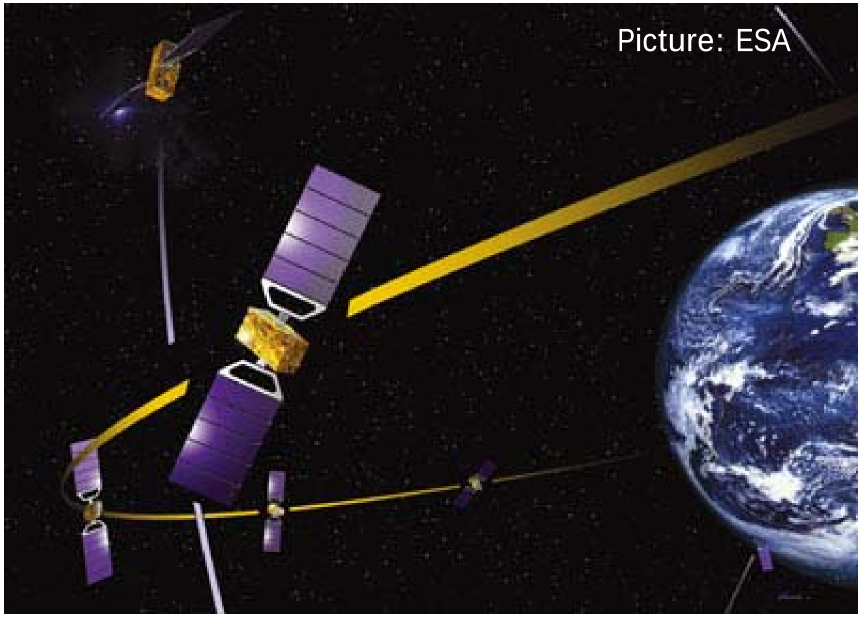
Satellite Navigation
Integrity and integer ambiguity resolution



AE4E08

Sandra Verhagen

Course 2010 – 2011, lecture 12

TU Delft

Today's topics

- Integrity and RAIM
- Integer Ambiguity Resolution

- Study Section 7.4 – 7.6 (not LMS algorithm in 7.5.3)

Integrity: performance measures

Integrity = ability of a system to provide timely warnings to users when the system should not be used

- HPE/VPE : Horizontal/Vertical Position Error
  (not known; due to measurement noise and biases)
- HPL/VPL : Horizontal/Vertical Protection Level
  horizontal/vertical position is assured to be within region defined by HPL/VPL
- HAL/VAL : Horizontal/Vertical Alarm Limit
  position error that should result in an alarm being raised
- TTA : Time To Alarm
  time between occurrence of integrity event (position error too large) and alarm being raised

Integrity: performance measures

- HPE/VPE : Horizontal/Vertical Position Error
  (not known; due to measurement noise and biases)
- HPL/VPL : Horizontal/Vertical Protection Level
  horizontal/vertical position is assured to be within region defined by HPL/VPL (can be calculated)
- HAL/VAL : Horizontal/Vertical Alarm Limit
  position error that should result in an alarm being raised

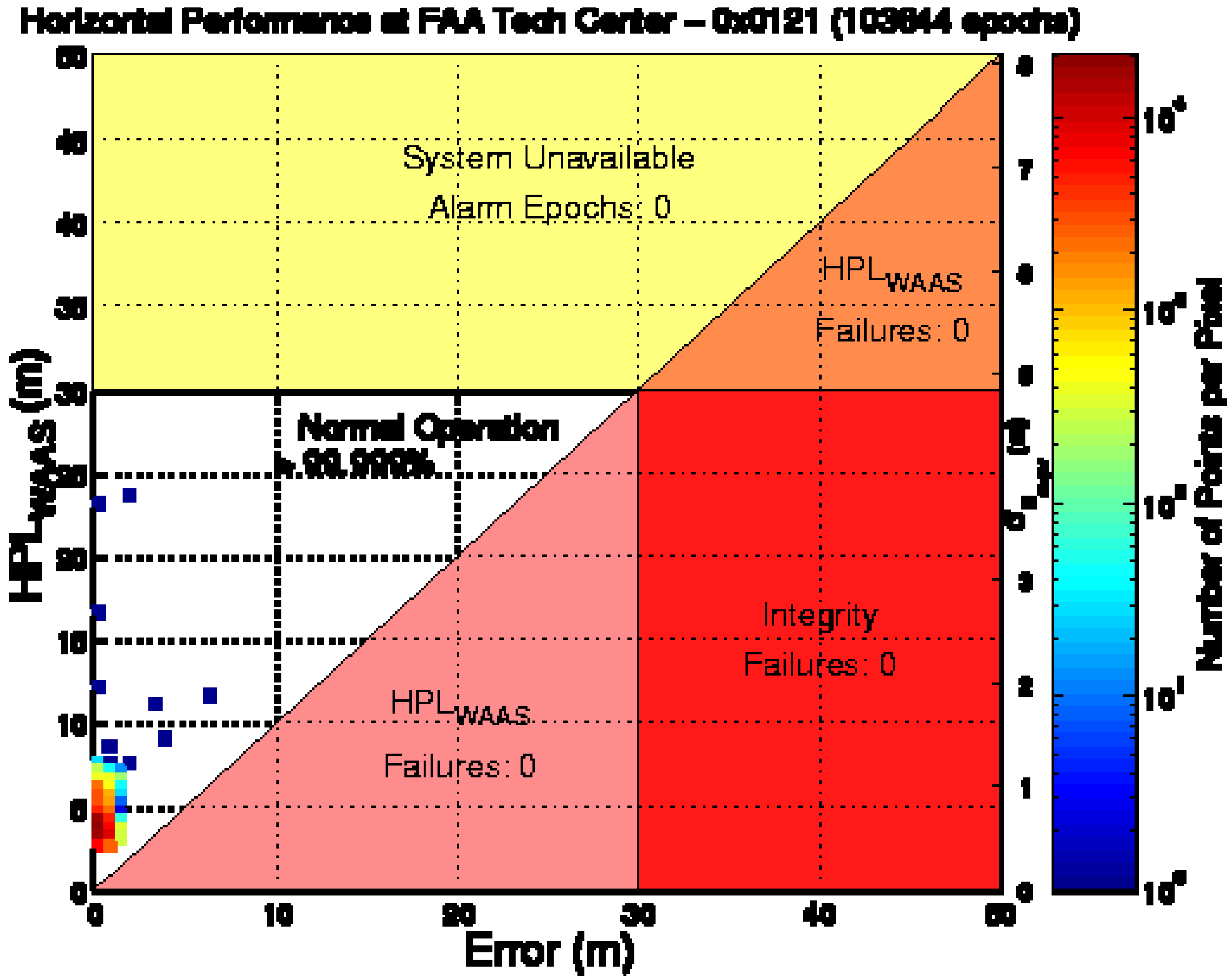
required: $P(\text{XPE} > \text{XPL}) <$ integrity risk

↓

$\text{XPL} > \text{XAL}$ → alarm, system unavailable

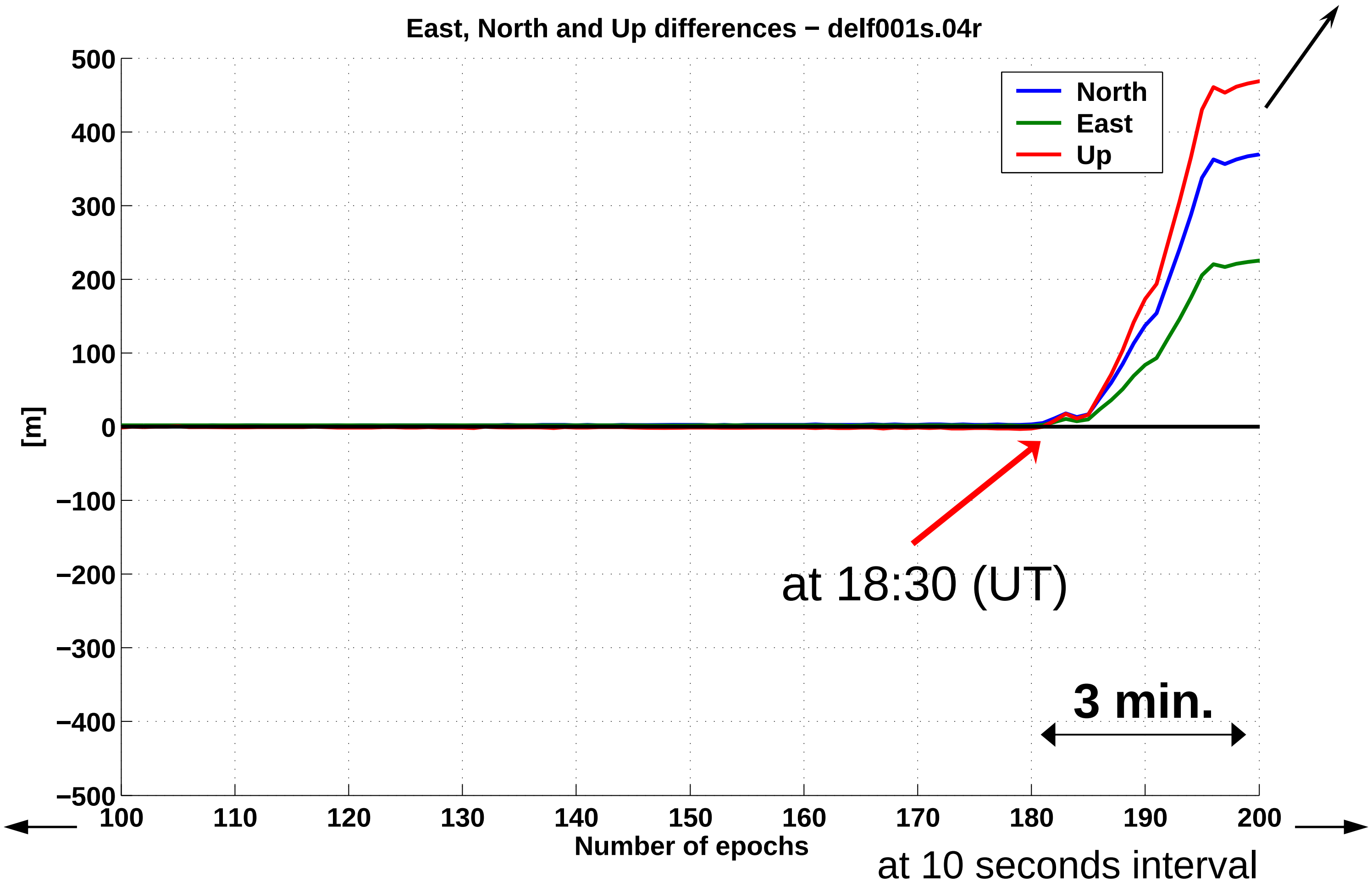
TU Delft

Stanford plots
http://waas.stanford.edu/metrics.html



TU Delft

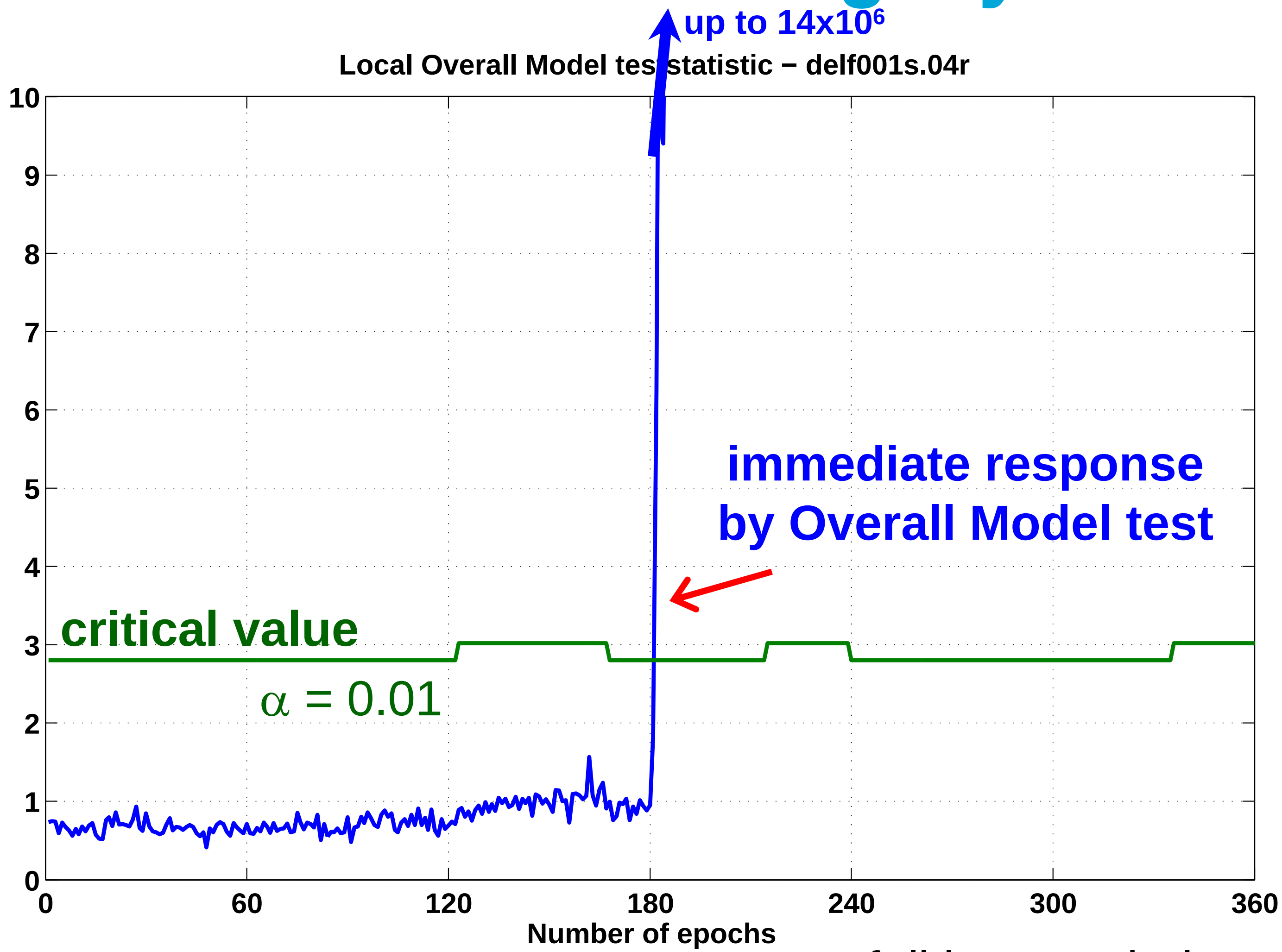
GPS PRN 23 Anomaly, 1 Jan, 2004



Not noticed by US for 3 hours
Picked up by EGNOS
Alternative: check @receiver

TU Delft

Receiver Autonomous Integrity monitoring



= full hour period

TU Delft

RAIM - Overall model test

RAIM: detect and correct for errors in GPS data @receiver

Overall model test: does H_o provide good model?

Model: $\underline{y} = Ax + \underline{e}$

Measurements: $y \rightarrow \quad \hat{x} = \left(A^T Q_{yy}^{-1} A\right)^{-1} A^T Q_{yy}^{-1} y$

$$\hat{y} = A\hat{x}$$

Residuals: $\hat{e} = y - \hat{y}$

"Mismatch": $\underline{T}_{q=m-n} = \underline{\hat{e}}_o^T Q_{yy}^{-1} \underline{\hat{e}}_o$

RAIM - Overall model test

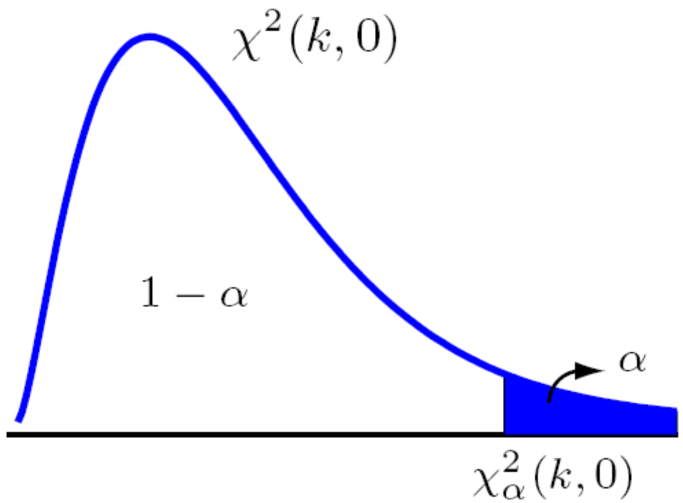
RAIM: detect and correct for errors in GPS data @receiver

Overall model test: does H_o provide good model?

$$\underline{T}_{q=m-n} = \hat{\underline{e}}_o^T Q_{yy}^{-1} \hat{\underline{e}}_o$$

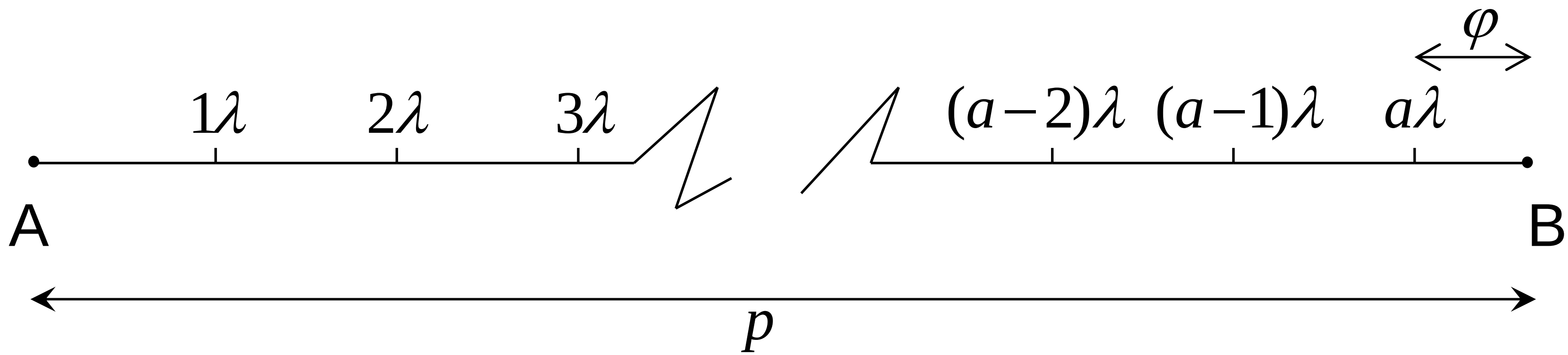
$$H_o : \ \underline{T}_q \sim \chi^2(m-n, 0)$$

$$H_a : \ \underline{y} \in \mathbb{R}^m$$



TU Delft

A Simple Ambiguity Resolution Example



ρ : unknown distance AB

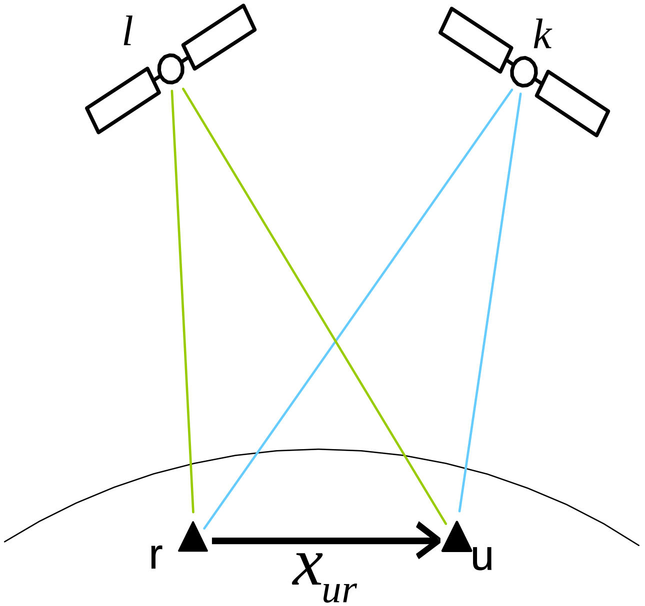
p : measured distance AB (σ_p = 20 cm)

φ : measured fraction distance AB (σ_φ = 2 mm)

a : number of times λ (= known) fits into ρ (a = unknown integer)

Model: $$E\begin{pmatrix}\varphi \\ p\end{pmatrix} = \begin{pmatrix}1 & -\lambda \\ 1 & 0\end{pmatrix}\begin{pmatrix}\rho \\ a\end{pmatrix}, \qquad D\begin{pmatrix}\varphi \\ p\end{pmatrix} = \begin{pmatrix}\sigma_\varphi^2 & 0 \\ 0 & \sigma_p^2\end{pmatrix}, \qquad a = \text{integer}$$

Relative Positioning: Double Differencing



- elimination of **receiver clock errors**
- elimination **of initial receiver phase offsets**
- **DD phase ambiguity** is an **integer** number!

DD code observation:

$$\rho_{ur,i}^{(kl)} = \left(-\mathbf{1}_r^{(kl)}\right)^T \mathbf{x}_{ur} + \mu_i I_{ur}^{(kl)} + T_{ur}^{(kl)} + \varepsilon_{\rho_i,ur}^{(kl)}$$

DD phase observation:

$$\Phi_{ur,i}^{(kl)} = \left(-\mathbf{1}_r^{(kl)}\right)^T \mathbf{x}_{ur} - \mu_i I_{ur}^{(kl)} + T_{ur}^{(kl)} + \lambda_i N_{ur,i}^{(kl)} + \varepsilon_{\Phi_i,ur}^{(kl)}$$

relative receiver position

integer DD ambiguity

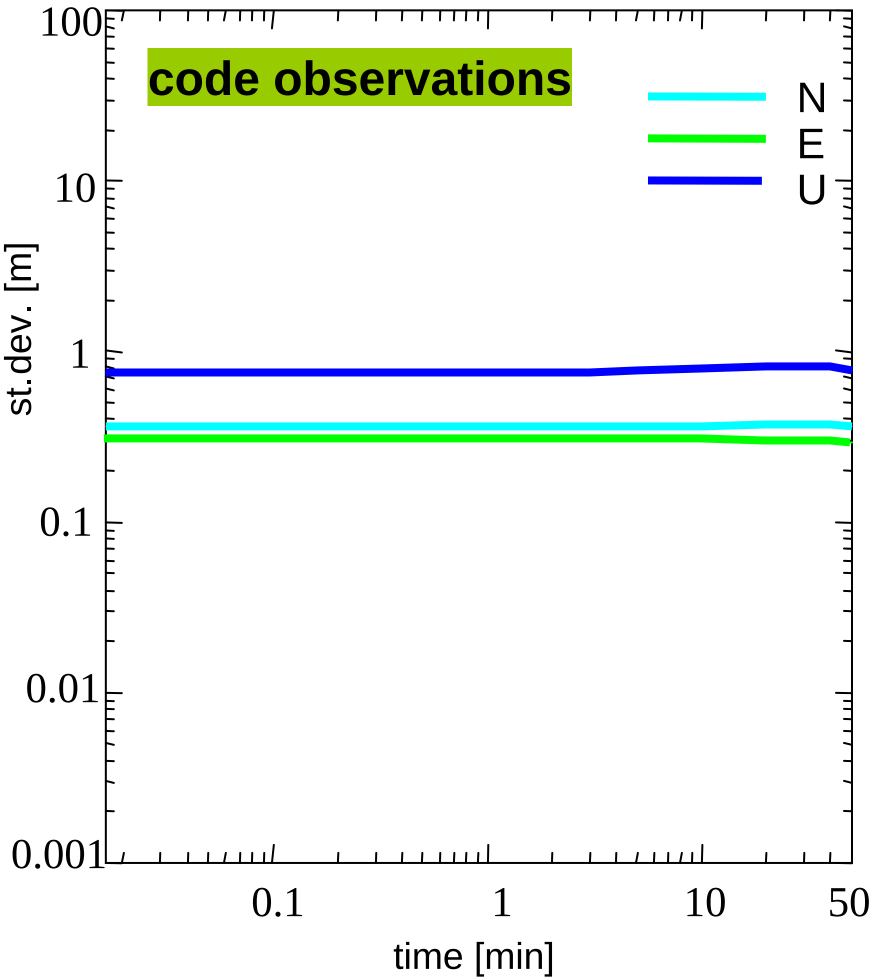
Resolution of the DD ambiguities

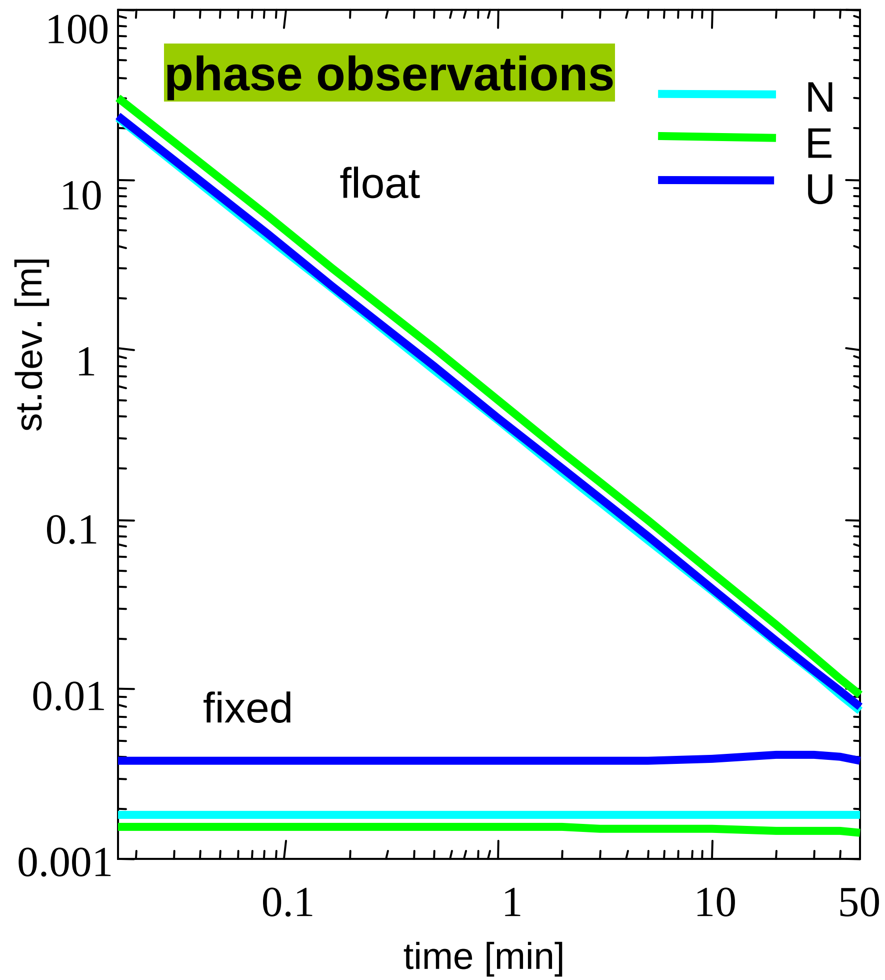
- code observation: **dm** precision
- phase observation: **mm** precision,
  - but: **receiver-satellite geometry** has to change considerably (long observation time) to solve position with mm-cm accuracy

  - if DD ambiguities are resolved to **integers within a short time (or instantaneously)**, positions (and other parameters) can be solved with mm-cm accuracy

Precision of relative GPS positioning

two epochs of data: varying time span

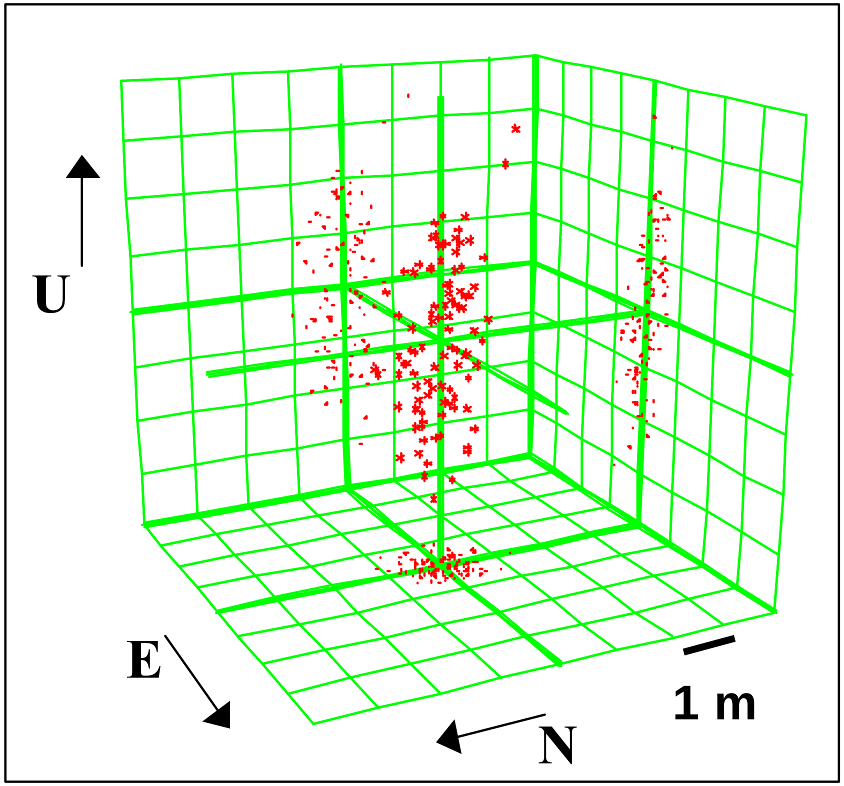




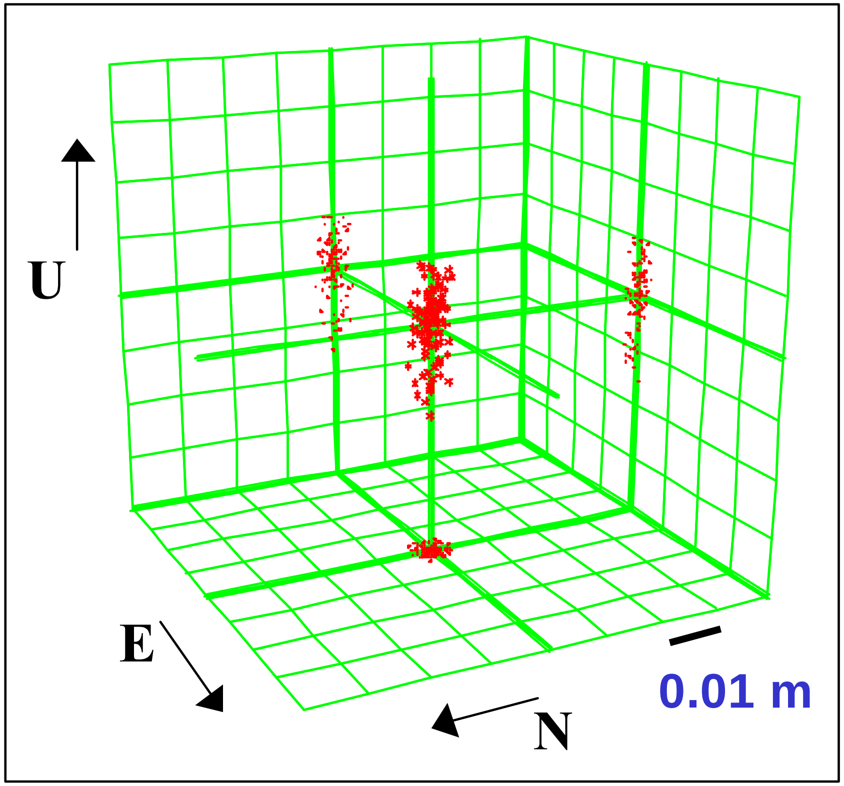
TU Delft

Precision code vs. phase observations

code observations



phase observations



- both RELATIVE positioning
- phase: provided that the integer ambiguity is KNOWN

Ionosphere-fixed, -float, -weighted model

- **Ionosphere-fixed model:**
  - **no** differential ionospheric delay parameters
  - observations may be **corrected** a priori for ionosphere
  - for **short** baselines only
  - can already be based on **single-frequency** data
- **Ionosphere-float model:**
  - **estimation** of differential ionospheric delays
  - no a priori corrections
  - for **long** baselines
  - based on at least **dual-frequency** data
- **Ionosphere-weighted model:**
  - **ionosphere corrections from network RTK 'subtracted'**
  - for **medium** to **long** baselines

GNSS model

In book:

$$\mathbf{y} = \mathbf{A}\mathbf{N} + \mathbf{G}\boldsymbol{\delta}\mathbf{x} + \boldsymbol{\varepsilon}$$

Observation equations:

$$\mathbf{y} = \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} + \mathbf{e}, \quad \mathbf{a} \in \mathbb{Z}^n; \qquad \mathbf{Q}_{\mathbf{yy}}$$

- $\mathbf{y}$ data vector
- $\mathbf{a}$ ambiguities
- $\mathbf{b}$ baseline coordinates & other unknowns
- $\mathbf{Q}_{\mathbf{yy}}$ variance-covariance matrix of data

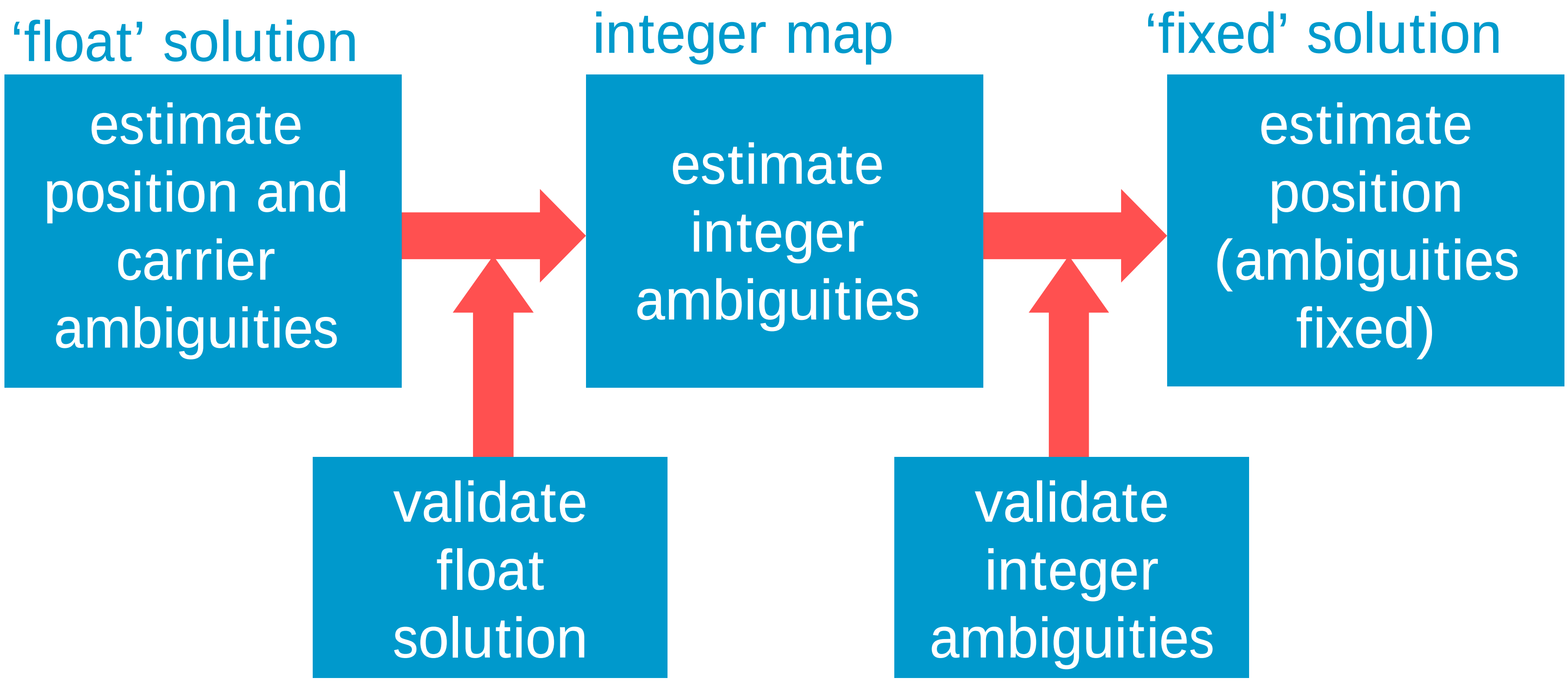
SUCCESSFUL INTEGER AMBIGUITY RESOLUTION

is the key to

FAST and PRECISE GNSS parameter estimation

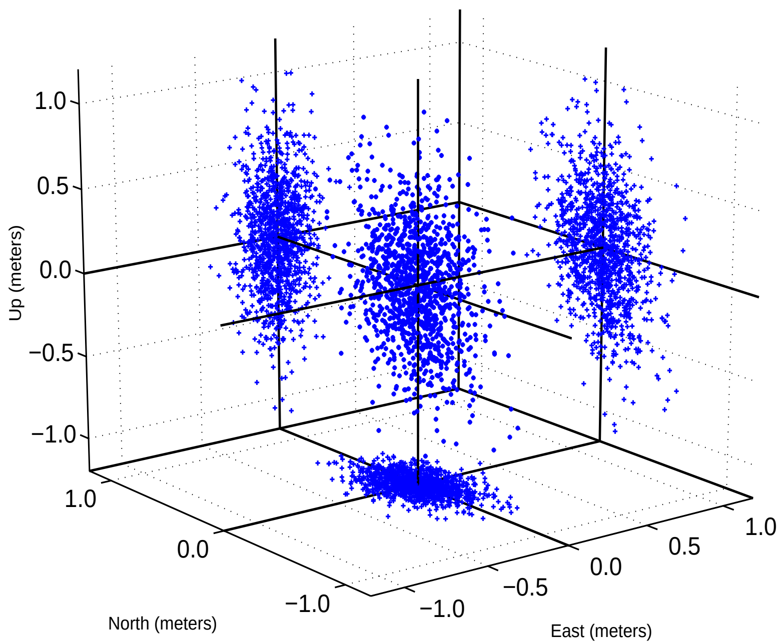
(baseline coordinates, attitude angles, orbit parameters, atmospheric delays)

Integer estimation

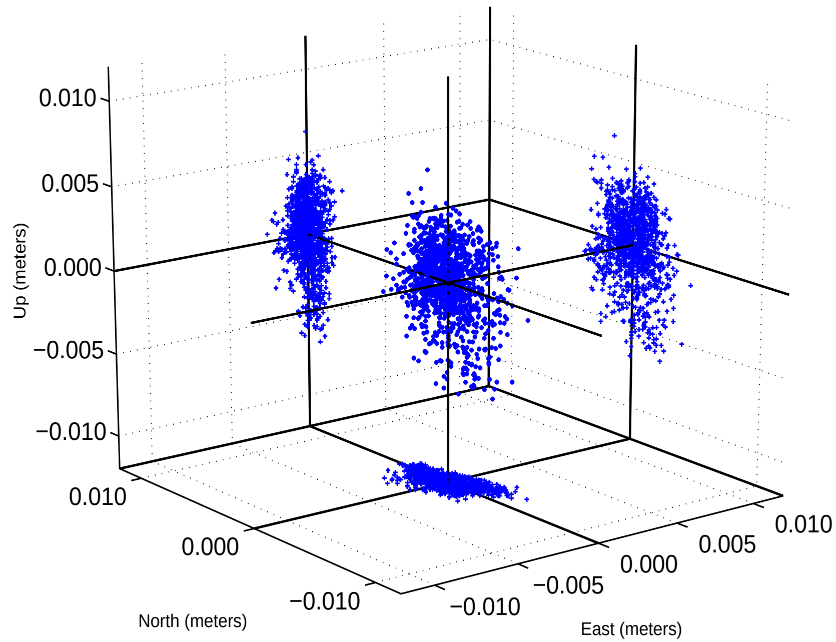


Float and fixed solution

Ambiguities not fixed



Ambiguities fixed

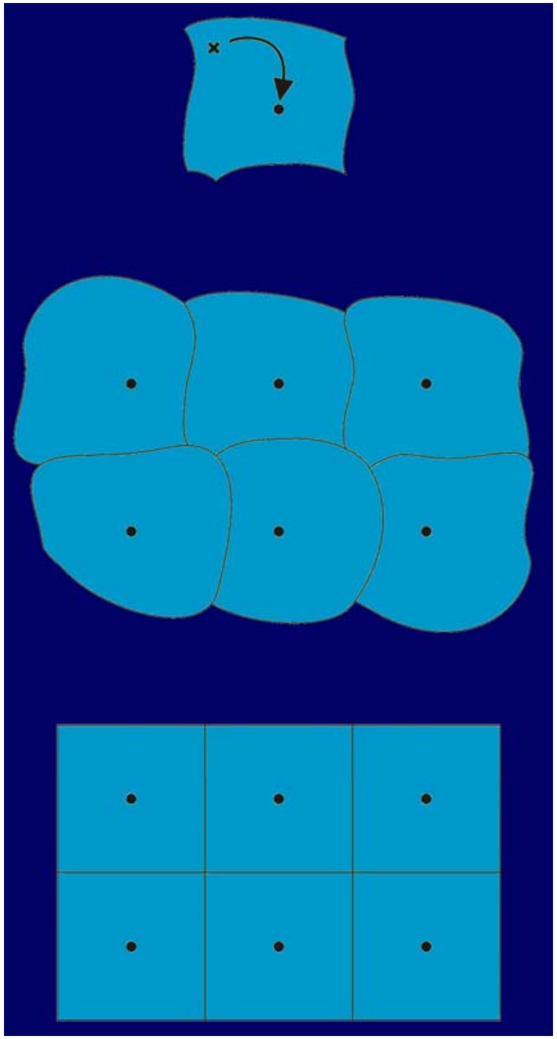


Integer estimation

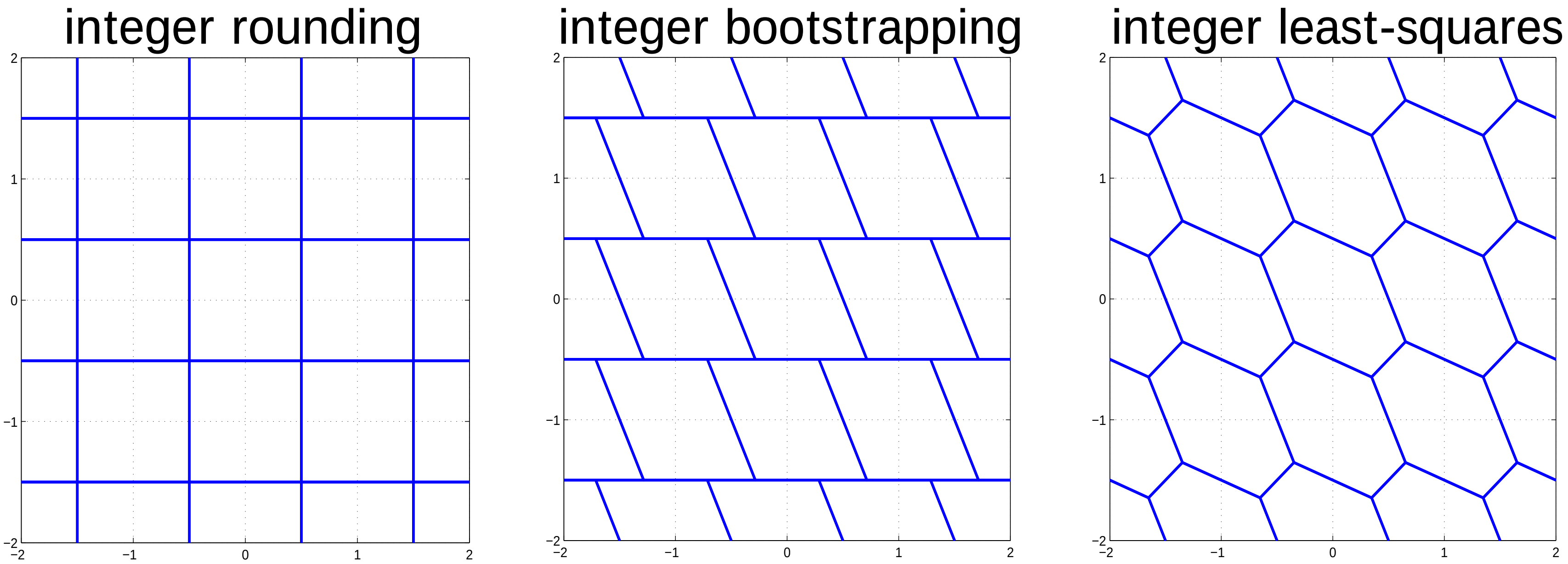
integer map $\hat{\mathbf{a}} \in \mathbb{R}^n \rightarrow S(\hat{\mathbf{a}}) = \breve{\mathbf{a}} \in \mathbb{Z}^n$

no holes & no overlap
→ there will always be **<u>ONE</u>** solution

translation invariant



Different choices of integer estimators



after their pull-in region

Ambiguity resolution

Integer ambiguities are derived from stochastic observations

Integer ambiguities are
not deterministic but ***stochastic***

input (stochastic)

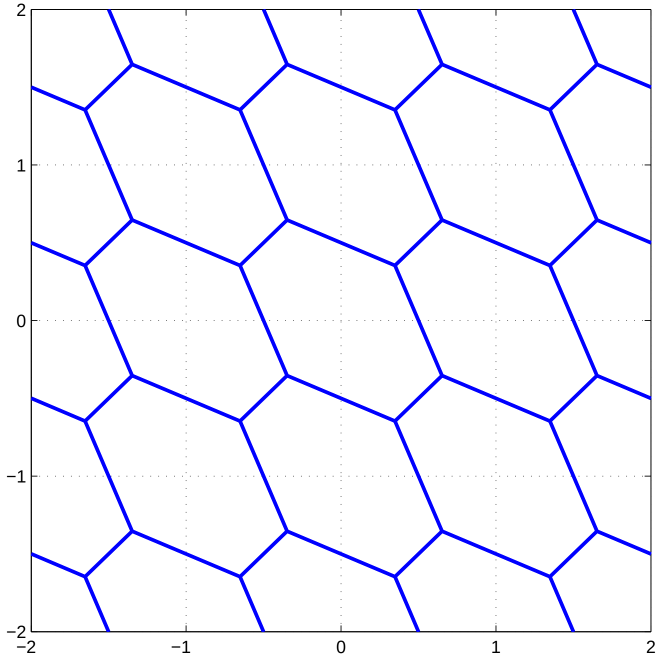
$$\hat{\mathbf{a}} \in \mathbb{R}^n \rightarrow S(\hat{\mathbf{a}}) = \breve{\mathbf{a}} \in \mathbb{Z}^n$$

output (stochastic)

Integer estimation

Optimal integer estimator: integer least-squares

$$\breve{\mathbf{a}} = \arg\min_{\mathbf{z}\in\mathbb{Z}^n} \left\| \hat{\mathbf{a}} - \mathbf{z} \right\|^2_{\mathbf{Q}_{\hat{\mathbf{a}}\hat{\mathbf{a}}}}$$



TU Delft

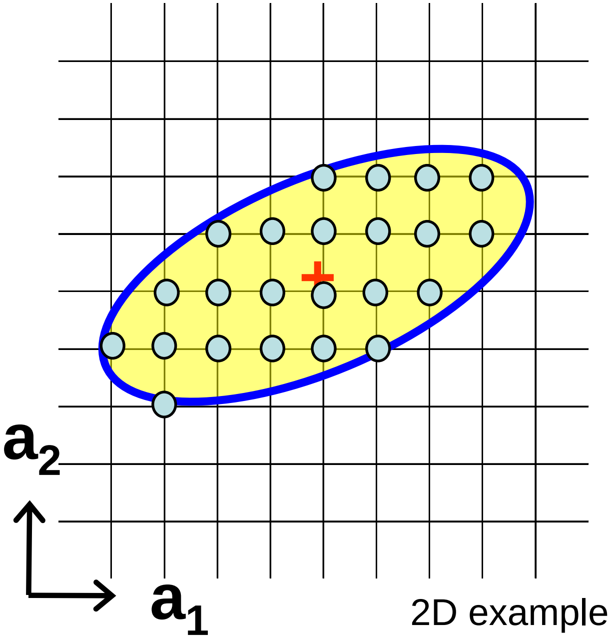
Ambiguity search space: a (hyper-) ellipsoid

- **centered** at $\hat{\mathbf{a}}$
- **shape** governed by $\mathbf{Q}_{\hat{a}\hat{a}}$
- find all integers $\mathbf{z}$ for which

$$(\hat{\mathbf{a}}-\mathbf{z})^T \mathbf{Q}_{\hat{a}\hat{a}}^{-1} (\hat{\mathbf{a}}-\mathbf{z}) \leq \chi^2$$

- χ^2 should be set such that search space contains at least **one** integer vector

- select the $\mathbf{z}$ which provides minimum

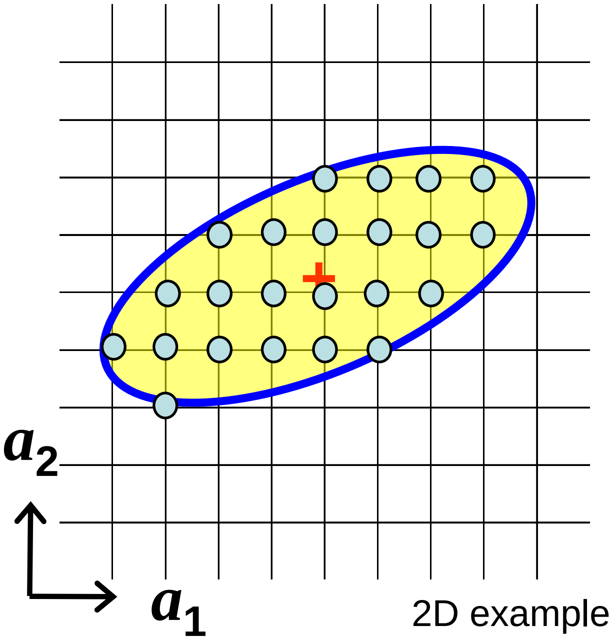


+ $\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2$ (float solution)

○ candidate integer solution

Integer ambiguity resolution

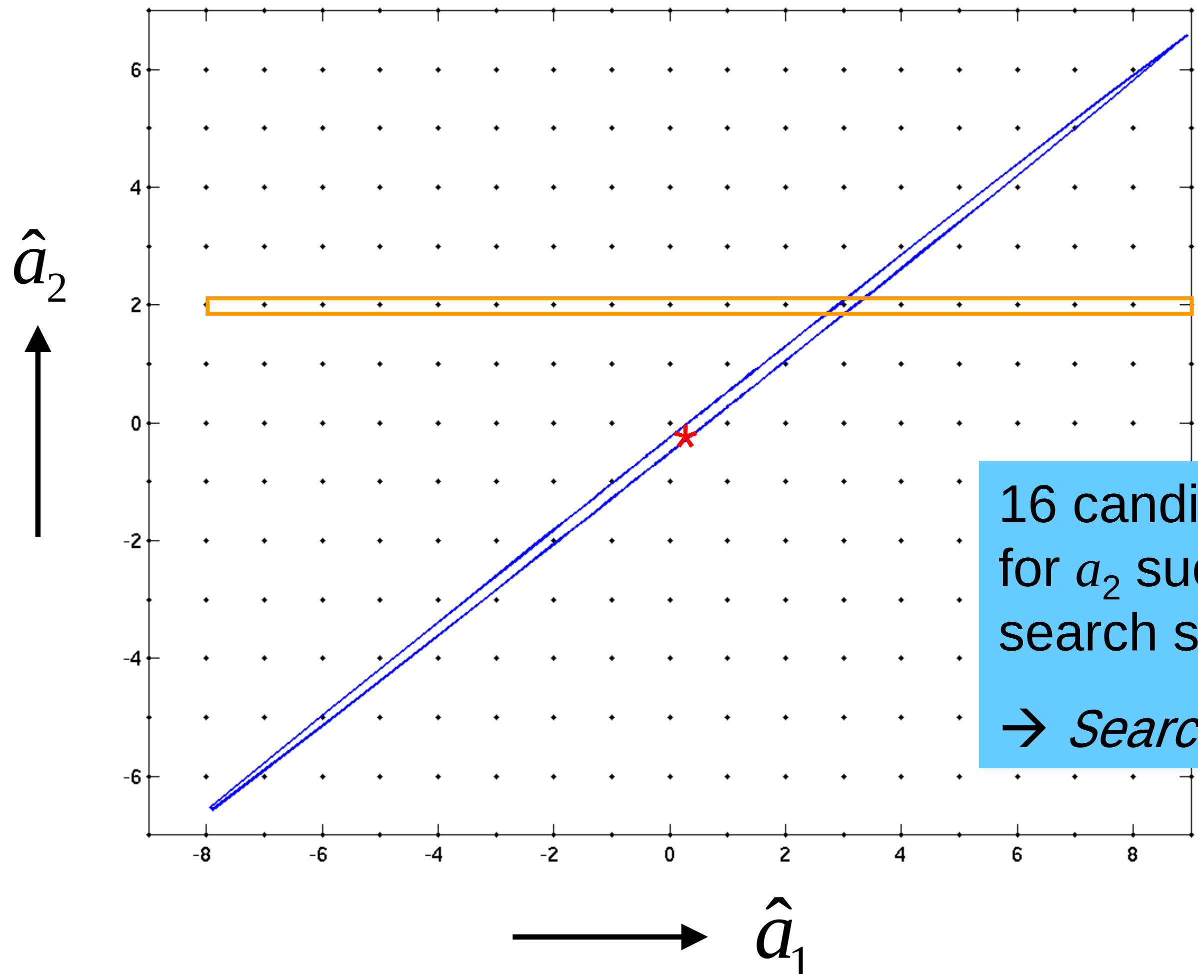
- Float solution: least-squares
- Integer search: find integer solution with shortest weighted distance to float solution (weighted by variance-covariance matrix of float ambiguites)
- Search difficult due to correlations
- LAMBDA: transformation of search space to make it efficient



2D example

Example: Ambiguity search space

Two dimensions, geometry-free, short baseline



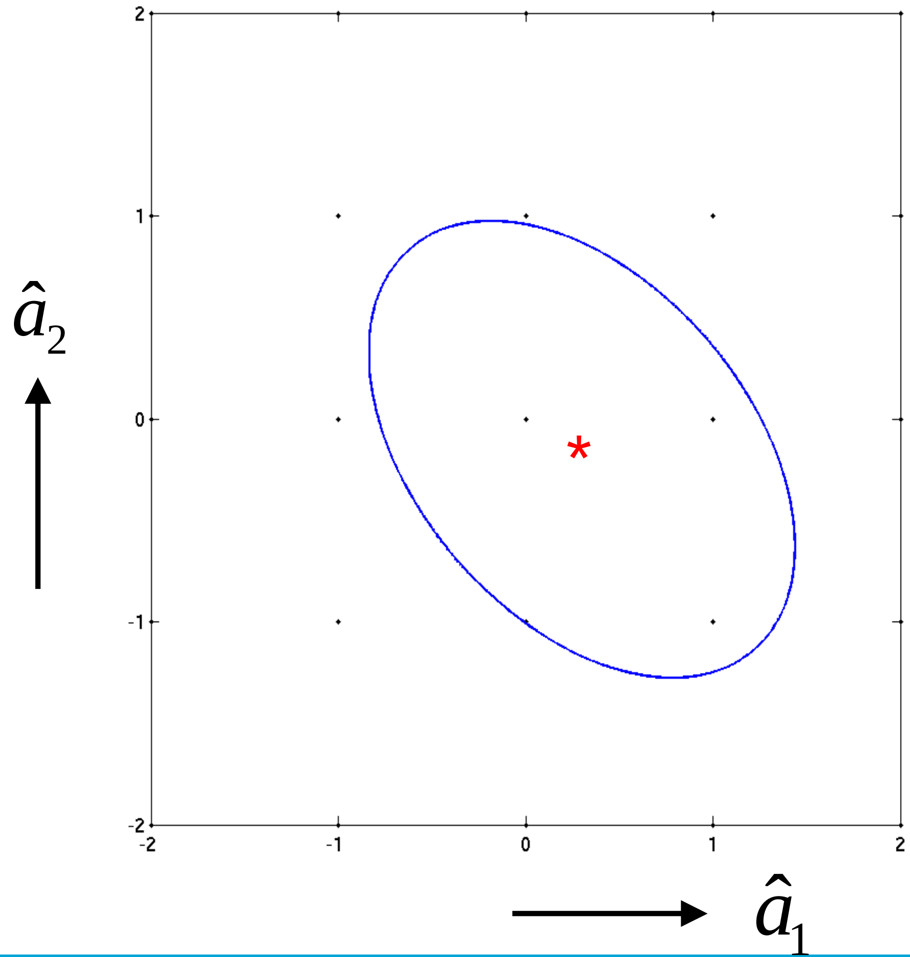
16 candidates for a_1, but only one for a_2 such that (a_1, a_2) inside search space

→ *Search is inefficient*

TUDelft

Example: Ambiguity search space

Two dimensions, geometry-free, short baseline



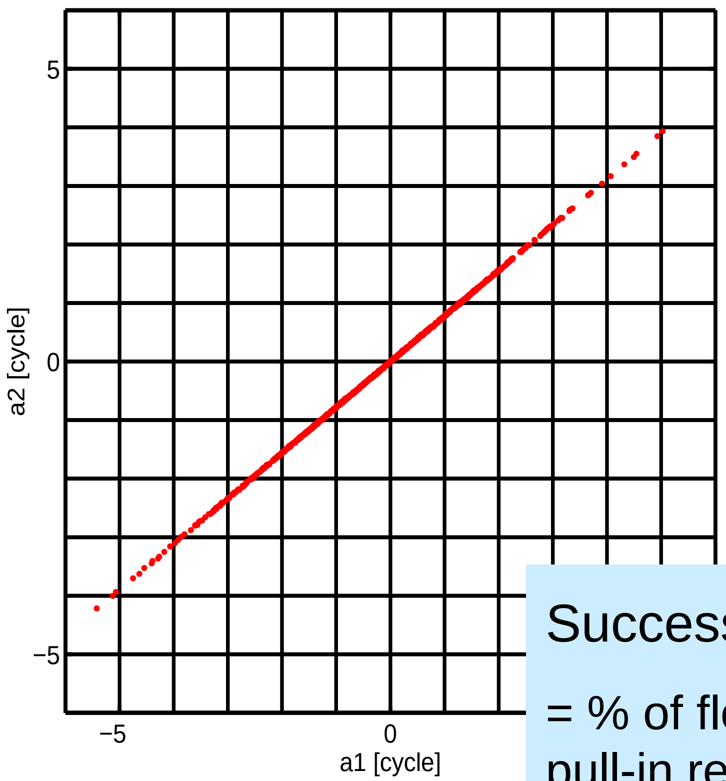
After decorrelation

Number of candidates INSIDE search space is same

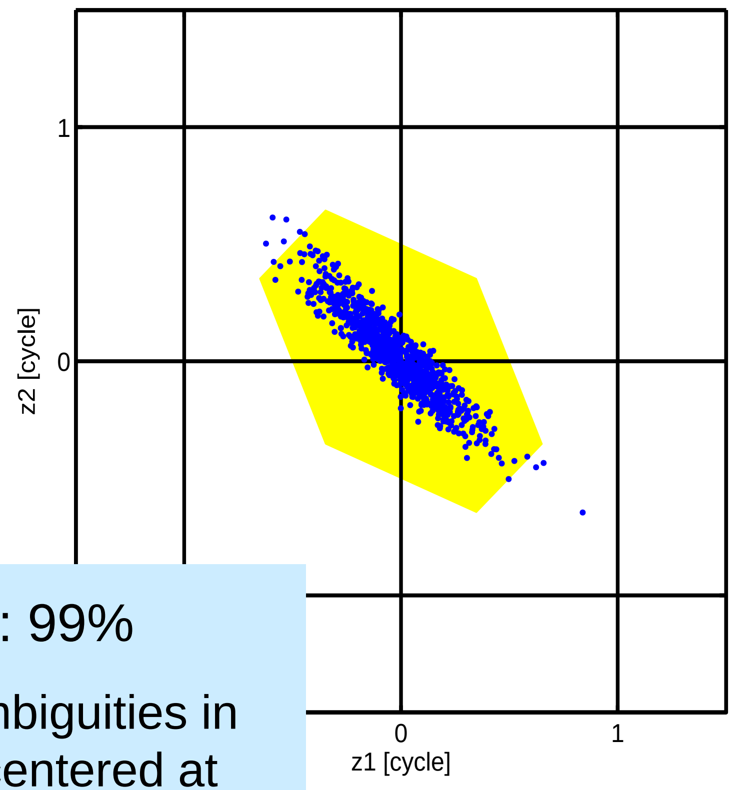
→ *Search is efficient*

Ambiguity estimation and success rate
Example based on real data (1000 epochs)

Distribution of original ambiguities



Distribution of transformed ambiguities



Success rate: 99%

= % of float ambiguities in pull-in region centered at correct integer value

Integer ambiguity resolution

Successful ambiguity resolution depends on precision of float solution, which depends on:

- baseline length (tropo + iono delays)
- satellite geometry
- precision of code and phase observations
- # frequencies

→ Change in satellite geometry helps (long duration)

LAMBDA method

Integer estimation:

- optimal : maximum success rate
- efficient : (near) real-time

LAMBDA

Applicable to wide variety of models

- With or without relative satellite-receiver geometry
- Stationary or moving receivers
- With or without atmospheric delays
- Single- or multi-baseline
- One, two, three or more frequencies (any GNSS)

LAMBDA

Baseline models

Parameters

	Geometry-free	Roving-receiver	Stationary-receiver
Ranges	N		
Station coordinates		N	C
Ambiguities	C	C	C
Ionospheric delays	N*)	N*)	N*)

N - New parameter introduced for each observation epoch

C - Constant parameter for entire observation period

*) - Long baselines only

Ambiguity Resolution Methods

- Search in the 3-dimensional position space (e.g. ambiguity function method); *Now deprecated*

- Linear combination of code and phase (using widelane/narrowlane combinations)
  - performance worse with AS
  - has been improved by LAMBDA: 2-dimensional ambiguity resolution/search problem

  → **Geometry-free model**
- Search in the *n*-dimensional ambiguity space

  → **Geometry-based model**

Summary and outlook

- We covered it all! (except for the applications)

Next:

Applications: your presentations

Exam preparation: check blackboard!