ESEO

Bayesian data fusion: an application to vehicle location

Kalman filtering applied to vehicle geolocation

22 Nov 2021

ESEO

I3 Data Sciences & Telecoms

David Bétaille

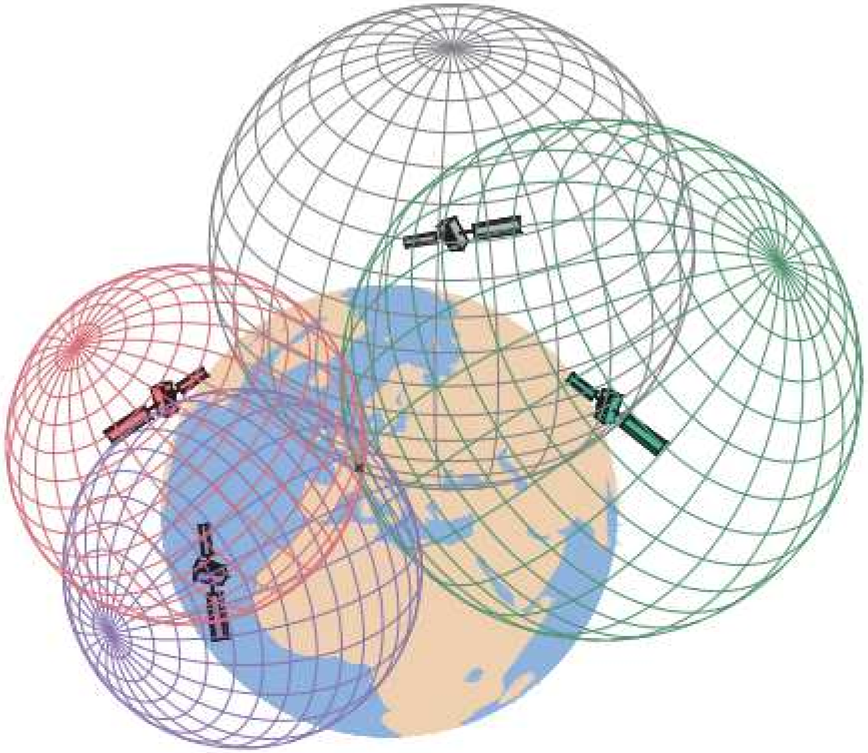
Univ. Gustave Eiffel

Contents

- Introduction
- GPS positioning
  - Standalone solution
  - DOP: Dilution Of Precision
  - Error propagation and bounding
- Coupling GPS + vehicle sensors & dynamic model
  - Bayesian filtering (Kalman)
  - Coupling modes: loose / tight coupling

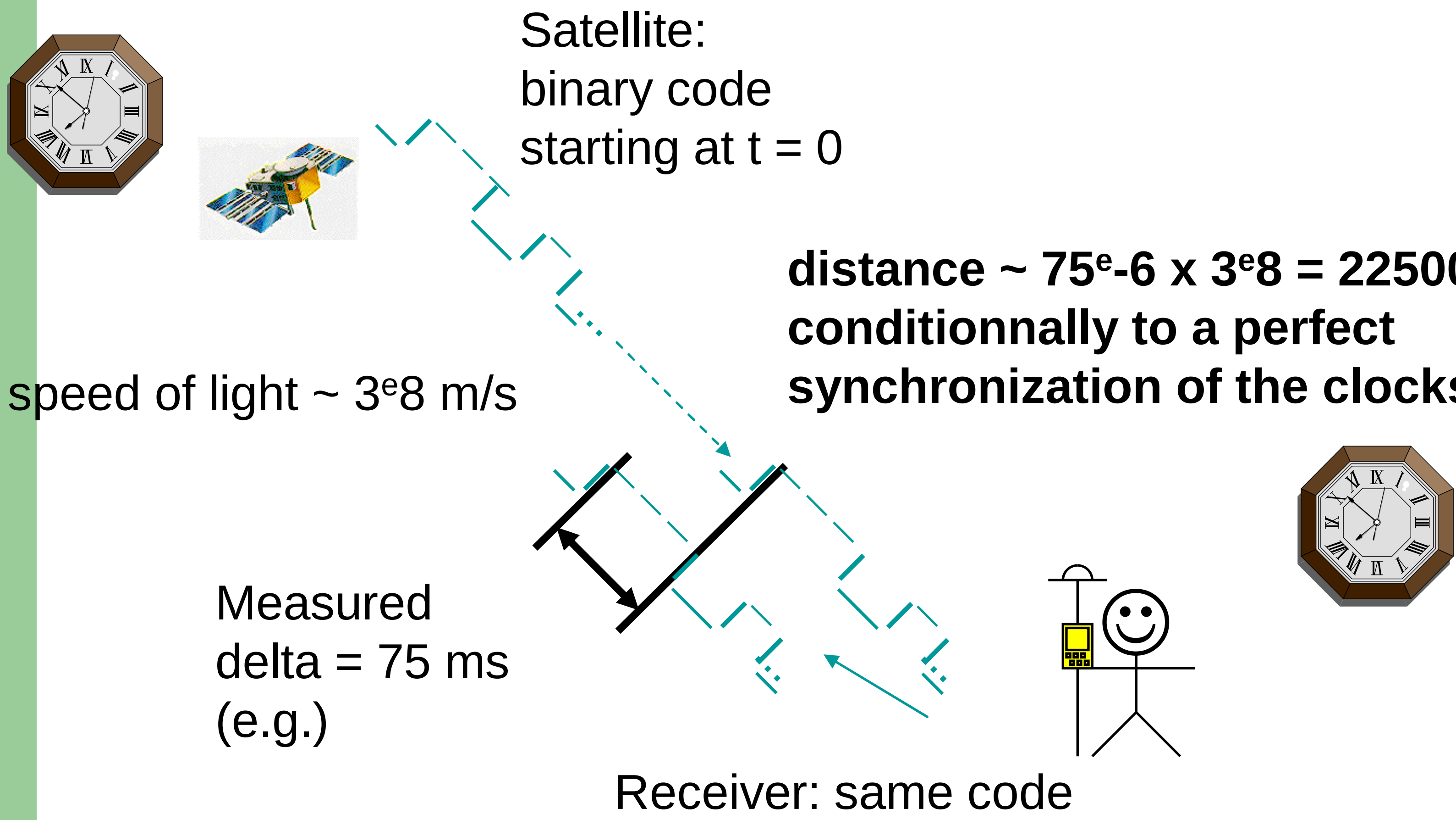
Trilateration

measuring distances
by travel time of satellite-
to-receiver signals



3D+rckt (∩ 4 spheres minimum)
=> solution

Measuring satellite-receiver distance



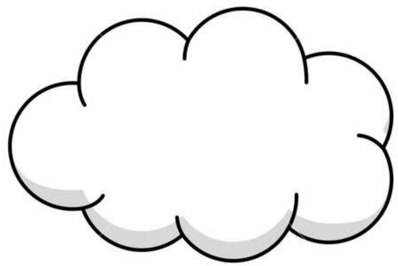
distance ~ 75ᵉ-6 x 3ᵉ8 = 22500 km conditionnally to a perfect synchronization of the clocks

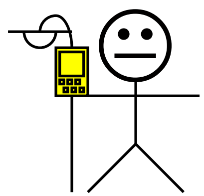
Main problems…

- clock offsets
- orbital errors

- atmosphere delays
iono and tropo
(min. 2m to resp. max. 20m and 50m)

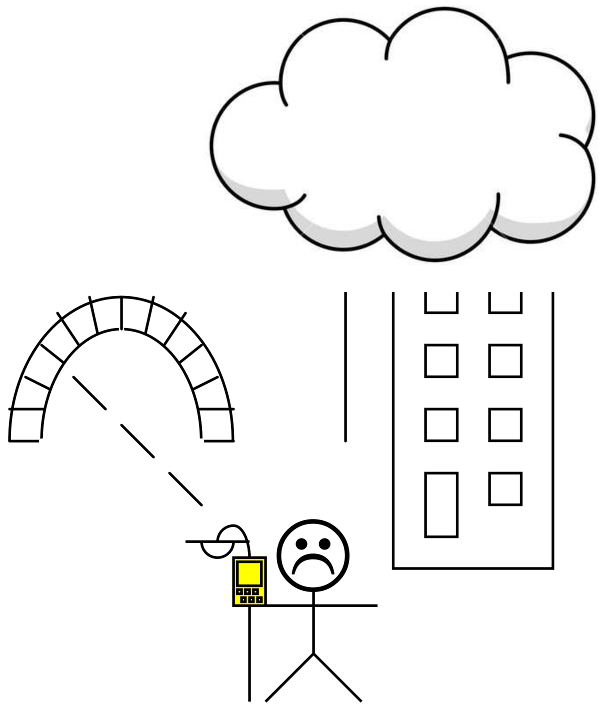
e.g. the iono GPS broadcast model and the Hopfield tropo model (with Essen and Froome coefficients and standard meteorological parameters)





Main problems…

- clock offsets
- orbital errors
- atmosphere delays
- low signal-to-noise ratio and **multipath** (reflection and diffraction locally at the receiver)



GPS positioning / code (1)

Observation equation and position of the problem

$\rho_j = R_j + c.dt$ **(1)** (c: speed of light)

- ρ_j: measured pseudo-range
- R_j: true geometric distance bw the receiver and the satellite j: $R_j = ||\mathbf{s_j} - \mathbf{u}||$ with

  $\mathbf{s_j} = (x_j\ y_j\ z_j)^T$ coordinates of the satellite j

  $\mathbf{u} = (x_u\ y_u\ z_u)^T$ coordinates of the receiver
- dt: receiver clock offset / GPS time

Note: one applies models for satellite clock offsets and atmospheric delays and one neglects other error terms (multipath, thermal noise…).

GPS positioning / code (2)

<u>Observation equation and position of the problem</u>

4 unknown ➔ 4 satellites are needed

The following system must be solved, with 4 equations and 4 unknown: x_u, y_u, z_u, dt j = 1 to 4:

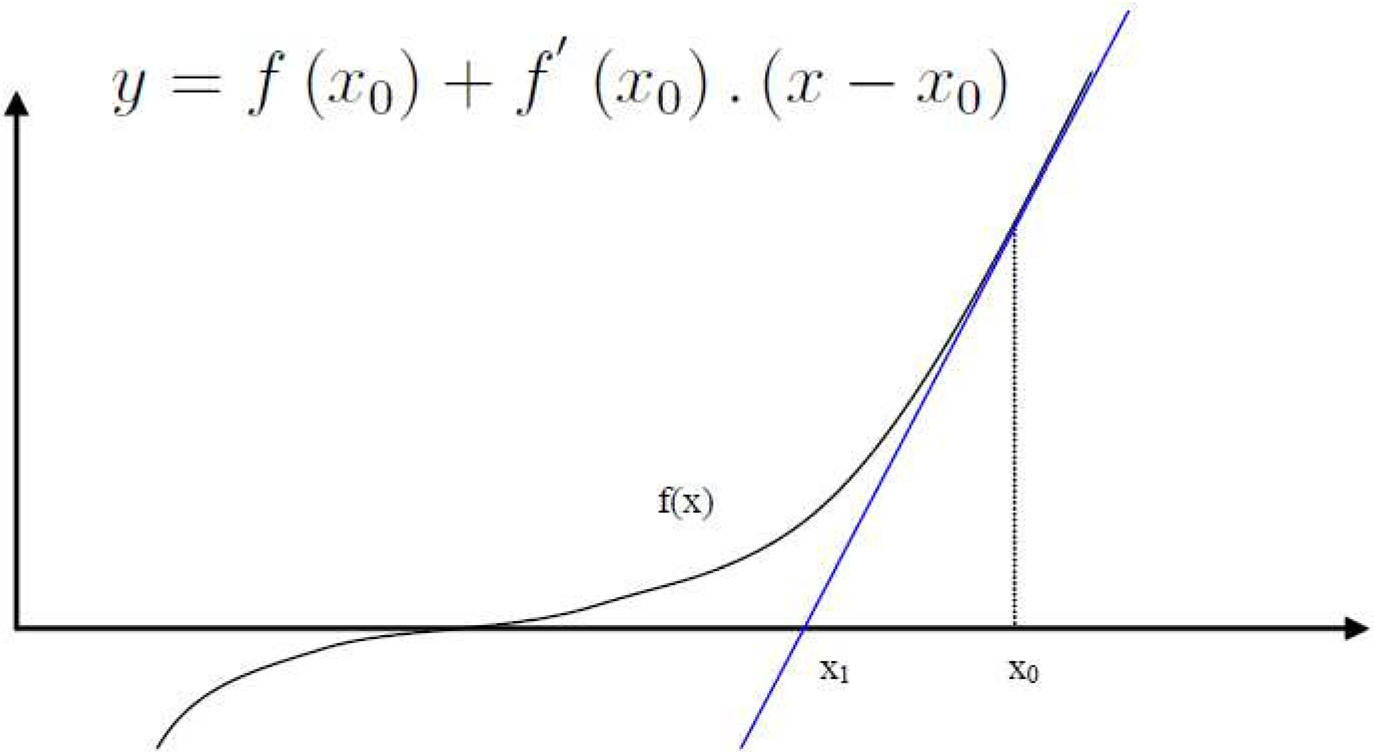
$$\rho_j = \sqrt{(x_j-x_u)^2 + (y_j-y_u)^2 + (z_j-z_u)^2} + c.dt$$

$$\rho_j = f_j(x_u, y_u, z_u, dt) \quad \textbf{(2)}$$

One can solve this system by **iterative linearization** and **least squares** (or by **Kalman filtering** under dynamic hypotheses), starting from an approximate solution

Iterative method (Newton-Raphson)

Like solving f(x)=0 with a scalar non linear fonction f



Linearization (1)

The function f is non linear and it should be linearized

- Let $\mathbf{u}° = (x°_u\ y°_u\ z°_u\ dt°)^T$ be an approximate solution
- One computes the approximate pseudo-ranges from this solution: $\rho°_j = f_j(x°_u, y°_u, z°_u, dt°)$ **(3)**

One notes:

$$x_u = x°_u + \Delta x_u$$
$$y_u = y°_u + \Delta y_u$$
$$z_u = z°_u + \Delta z_u$$
$$dt = dt° + \Delta dt$$

- Linearization of **(2)** around the approximate solution:

$f_j(x_u, y_u, z_u, dt) = f_j(x°_u + \Delta x_u\ ,\ y°_u + \Delta y_u\ ,\ z°_u + \Delta z_u\ ,\ dt° + \Delta dt)$ **(4)**

Linearization (2)

- 1st order Taylor development:

$$f_j(x_u, y_u, z_u, dt) = f_j(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ) + \frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial x'_u}\Delta x_u + \frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial y'_u}\Delta y_u + \frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial z'_u}\Delta z_u + \frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial dt'}\Delta dt \qquad (5)$$

- Derivatives of (2) are:

$$\frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial x_u} = -\frac{x_j - x^\circ_u}{\rho^\circ_j}$$

$$\frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial y_u} = -\frac{y_j - y'_u}{\rho^\circ_j}$$

$$\frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial z_u} = -\frac{z_j - z^\circ_u}{\rho^\circ_j}$$

and

$$\frac{\partial f(x^\circ_u, y^\circ_u, z^\circ_u, dt^\circ)}{\partial dt} = c$$

with: $\rho^\circ_j = \sqrt{(x_j - x^\circ_u)^2 + (y_j - y^\circ_u)^2 + (z_j - z^\circ_u)^2}$

Linearization (3)

- Reporting into **(5)**, one gets:

$$\rho_j = \rho^\circ{}_j - \frac{x_j - x^\circ{}_u}{\rho^\circ{}_j}\Delta x_u - \frac{y_j - y^\circ{}_u}{\rho^\circ{}_j}\Delta y_u - \frac{z_j - z^\circ{}_u}{\rho^\circ{}_j}\Delta z_u + c\Delta dt \qquad \textbf{(6)}$$

The observation equation **(2)** has been linearized in function of the unknown Δx_u, Δy_u, Δz_u and Δdt

(Note: keep in mind that $\rho^\circ{}_j$ is known and ρ_j is measured)

Linearization (4)

- Denoting: $\Delta\rho_j = \rho_j - \rho^\circ{}_j$ and: $a_{xj} = -(x_j - x^\circ{}_u) / \rho^\circ{}_j$

$$a_{yj} = -(y_j - y^\circ{}_u) / \rho^\circ{}_j$$

$$a_{zj} = -(z_j - z^\circ{}_u) / \rho^\circ{}_j$$

$\rho^\circ{}_j$ known and ρ_j measured

a_{xj}, a_{yj}, a_{zj} : cosine of unit vector $\mathbf{a_j}$ pointing satellite j from the approximate solution

equation (6) becomes:

$$\Delta\rho_j = a_{xj}\Delta x_u + a_{yj}\Delta y_u + a_{zj}\Delta z_u + c\Delta dt \qquad (7)$$

GPS positioning / code (3)

Inversion of the linearized system

- With:

$$\Delta\rho = \begin{bmatrix} \Delta\rho_1 \\ \Delta\rho_2 \\ \Delta\rho_3 \\ \Delta\rho_4 \end{bmatrix} \qquad H = \begin{bmatrix} a_{x1} & a_{y1} & a_{z1} & 1 \\ a_{x2} & a_{y2} & a_{z2} & 1 \\ a_{x3} & a_{y3} & a_{z3} & 1 \\ a_{x4} & a_{y4} & a_{z4} & 1 \end{bmatrix} \qquad \Delta x = \begin{bmatrix} \Delta xu \\ \Delta yu \\ \Delta zu \\ c\Delta dt \end{bmatrix}$$

one finally gets the following matrix equation:

$\Delta\rho = H\,\Delta\mathbf{x}$ **(8)** solved by: $\Delta\mathbf{x} = H^{-1}\,\Delta\rho$ **(9)**

- One iterates this process until the norm of vector $\Delta\rho$ gets below a certain limit

GPS positioning / code (4)

General case (N > 4 satellites) : over-determination

The solution $\Delta \mathbf{x} = H^{-1} \Delta\rho$ exists in case the problem is exactly determined (in this case, H is a 4 x 4 matrix)

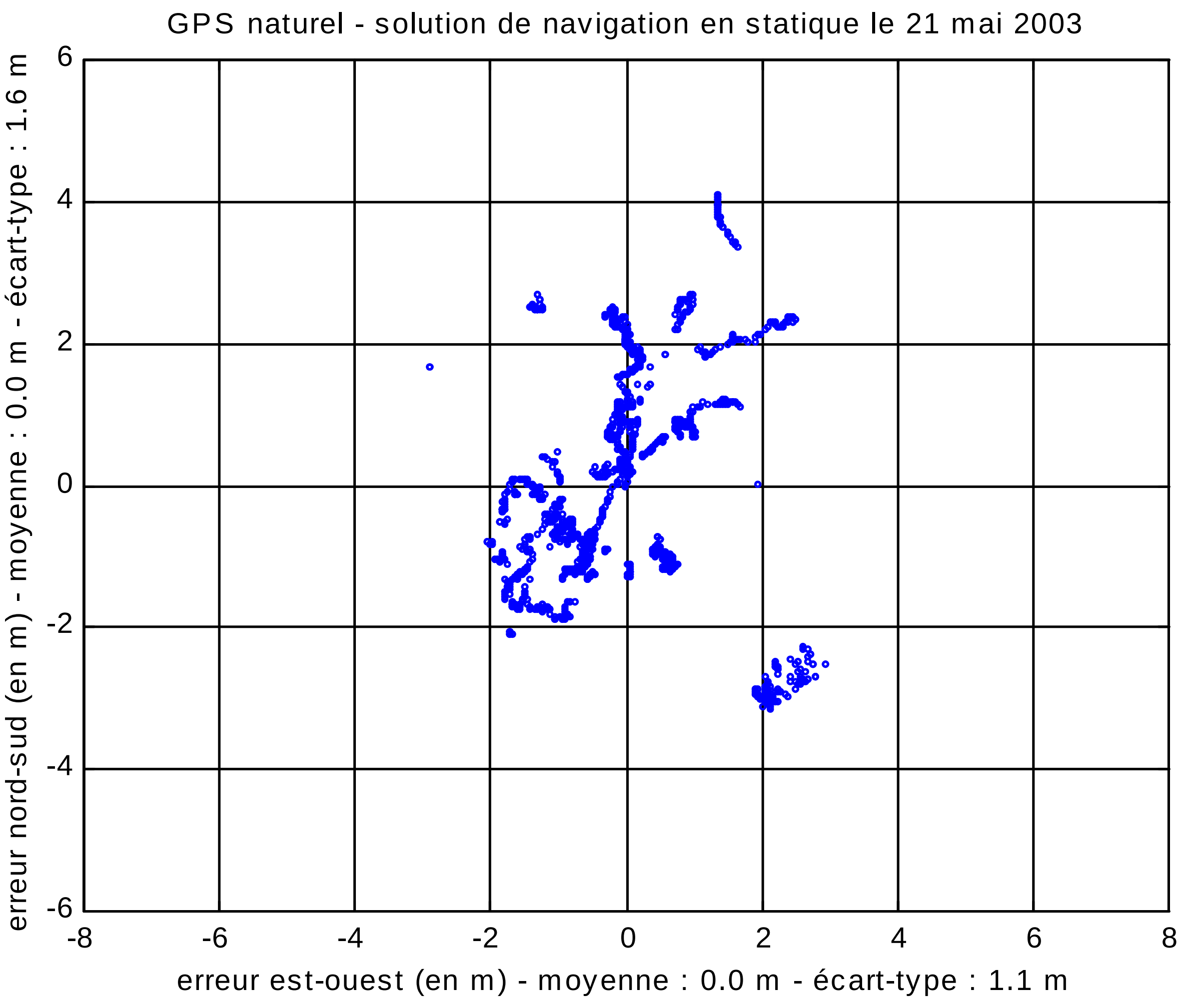
In case more than 4 satellites are observed, the problem is over-determined and equation **(8)** least-squares solution is obtained multiplying its 2 sides by H^T, then again by $(H^TH)^{-1}$: this is the pseudo-inverse or generalized inverse of H:

$$\Delta \mathbf{x} = (H^TH)^{-1} H^T \Delta\rho \quad \mathbf{(10)}$$

Note: one demonstrates that this solution minimizes the sum of the squared residuals (so called least-squares solution)

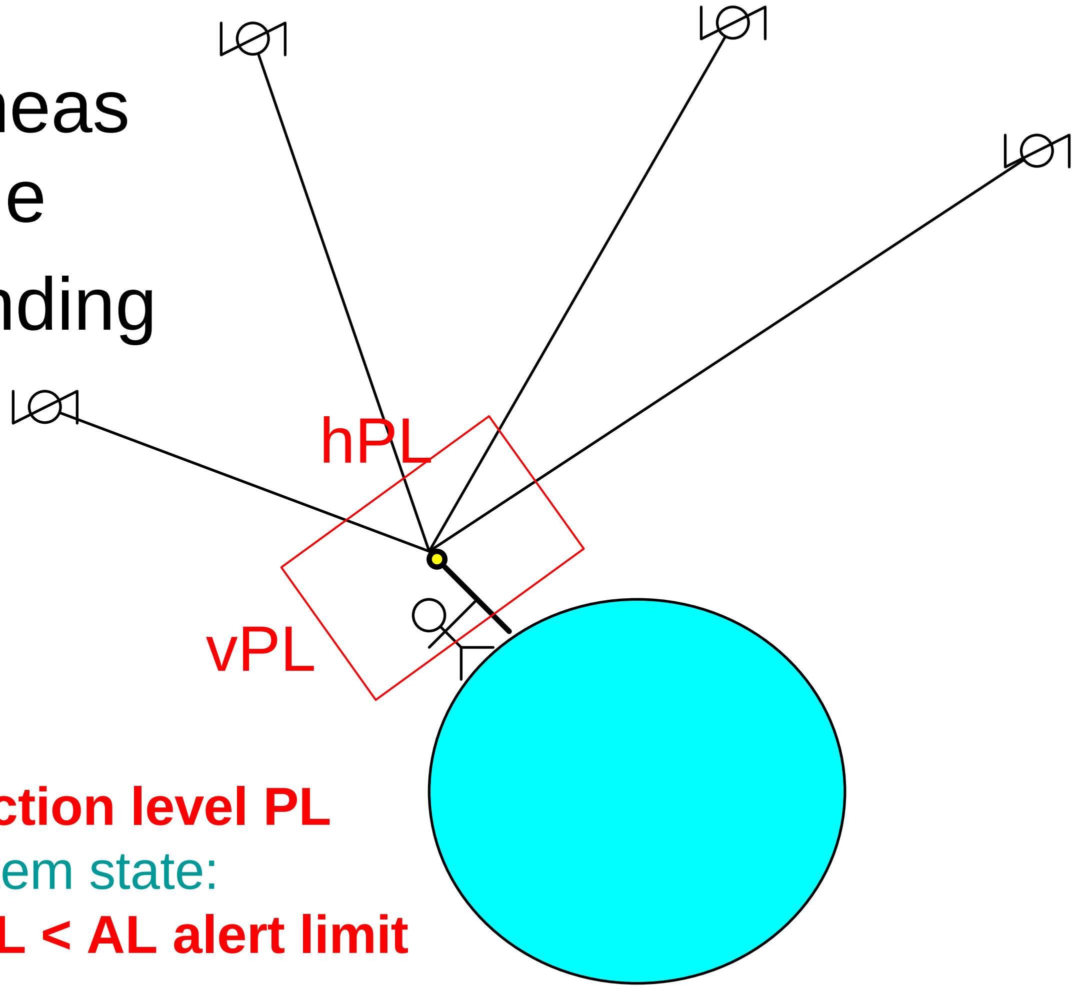
Standard GPS positioning service

Diagram of the horizontal error typical of the GPS standard service, for 24h. Observations collected near Nantes, in may 2003



Error propagation and bounding

- propagation from meas
  to position and time
- objective: error bounding



key concept: **protection level PL**
for secure system state:
position error PE < PL < AL alert limit

DOP dilution of precision

<u>Error propagation</u>

The H matrix (Jacobian matrix) is very important because it determines the **propagation of range errors**

$$H = \begin{bmatrix} a_{x1} & a_{y1} & a_{z1} & 1 \\ a_{x2} & a_{y2} & a_{z2} & 1 \\ a_{x3} & a_{y3} & a_{z3} & 1 \\ \dots & \dots & \dots & \dots \\ a_{xN} & a_{yN} & a_{zN} & 1 \end{bmatrix}$$

Let **d**ρ be the errors on pseudo-ranges

They propagate along x, y, z and t directions applying:

$\mathbf{dx} = H^{-1}\,\mathbf{d}\rho$ (or $\mathbf{dx} = (H^T H)^{-1} H^T\,\mathbf{d}\rho$) **(11) (or 12)**

dx: errors on the receiver position and clock term

DOP dilution of precision

In reality, the error vector **d**ρ is random and in practice its componants are assumed to be Gaussian with zero mean

The geometry being kwnown, **dx** is also Gaussian with zero mean (on long intervals, particularly for 24 h)

Let us compute the covariance matrix of **dx**

Definition: $\mathrm{cov}(\mathbf{dx}) = E(\mathbf{dx}\ \mathbf{dx}^T)$

Using **(11 or 12)**, one computes: $\mathrm{cov}(\mathbf{dx}) = E(H^{-1}\ \mathbf{d}\rho\ \mathbf{d}\rho^T\ H^{-1T})$

$$\mathrm{cov}(\mathbf{dx}) = (H^T H)^{-1} H^T \mathrm{cov}(\mathbf{d}\rho)\ H\ ((H^T H)^{-1})^T \qquad \mathbf{(13)}$$

DOP dilution of precision

Hypothesis: range errors *a priori* are independent and identically distributed, with a variance equal to σ^2_{UERE}:

$\mathrm{cov}(\mathbf{d}\rho) = I_{NxN}\ \sigma^2_{UERE}$ **(14)**

$\boldsymbol{\sigma^2_{UERE}}$ is named *User Equivalent Range Error variance* and it is equal to the quadratic sum of all the error sources

Using **(14)** in **(13)**, one gets:

$\mathrm{cov}(\mathbf{dx}) = (H^T\ H)^{-1}\ \sigma^2_{UERE}$ **(15)**

$\mathrm{cov}(\mathbf{dx}) = Q\ \sigma^2_{UERE}$

DOP dilution of precision

Q, **after XYZ to ENU rotation**, is used to estimate the errors on each dimension (x_u,y_u,z_u,t):

$Q = R\,(H^T H)^{-1} R^{-1}$

$$R = \begin{bmatrix} -\sin lon & \cos lon & 0 \\ -\sin lat \cos lon & -\sin lat \sin lon & \cos lat \\ \cos lat \cos lon & \cos lat \sin lon & \sin lat \end{bmatrix}$$

$$Q_x = \begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ D_{21} & D_{22} & D_{23} & D_{24} \\ D_{31} & D_{32} & D_{33} & D_{34} \\ D_{41} & D_{42} & D_{43} & D_{44} \end{bmatrix} \quad \mathrm{cov}(dx) = \begin{bmatrix} \sigma^2{}_{x_u} & \sigma^2{}_{x_u y_u} & \sigma^2{}_{x_u z_u} & \sigma^2{}_{x_u cdt} \\ \sigma^2{}_{x_u y_u} & \sigma^2{}_{y_u} & \sigma^2{}_{y_u z_u} & \sigma^2{}_{y_u cdt} \\ \sigma^2{}_{x_u z_u} & \sigma^2{}_{y_u z_u} & \sigma^2{}_{zu} & \sigma^2{}_{z_u cdt} \\ \sigma^2{}_{x_u cdt} & \sigma^2{}_{y_u cdt} & \sigma^2{}_{z_u cdt} & \sigma^2{}_{cdt} \end{bmatrix}$$

DOP dilution of precision

***k*DOP coefficients (1)**

1. for **all geometric parameters**:

Geometric Dilution of Precision

GDOP $=(D_{11}+D_{22}+D_{33}+D_{44})^{1/2}$, sqrt of the trace of Q_x

GDOP $= (\sigma_{xu}^2 + \sigma_{yu}^2 + \sigma_{zu}^2 + \sigma_{cdt}^2)^{1/2} / \sigma_{UERE}$

2. for **3D position**: *Position Dilution of Precision*

PDOP $=(D_{11}+D_{22}+D_{33})^{1/2}$

PDOP $= (\sigma_{xu}^2 + \sigma_{yu}^2 + \sigma_{zu}^2)^{1/2} / \sigma_{UERE}$

Notion de DOP dilution of precision

*k*DOP coefficients (2)

3. for **horizontal position**:

Horizontal Dilution of Precision

HDOP = $(D_{11}+D_{22})^{1/2}$

HDOP = $(\sigma_{xu}^2 + \sigma_{yu}^2)^{1/2} / \sigma_{UERE}$

4. for **height** : *Vertical Dilution of Precision*

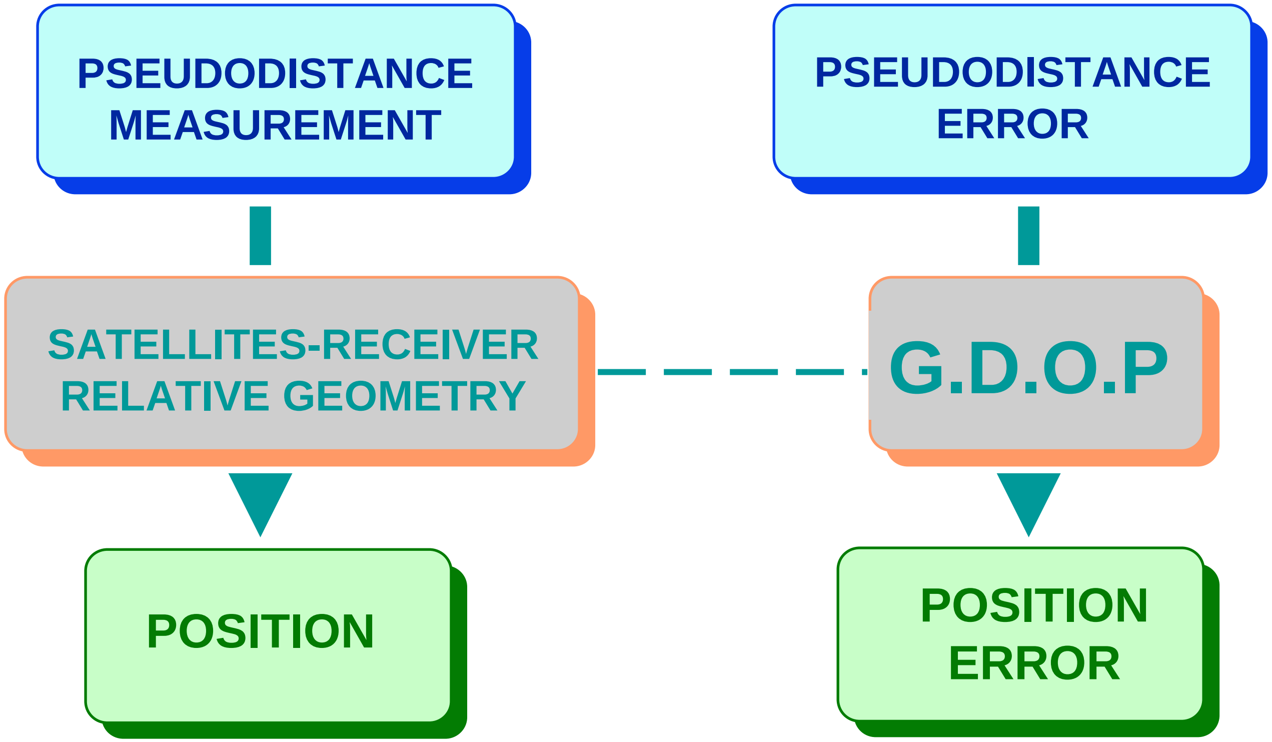
VDOP = $(D_{33})^{1/2}$

VDOP = $\sigma_{zu} / \sigma_{UERE}$

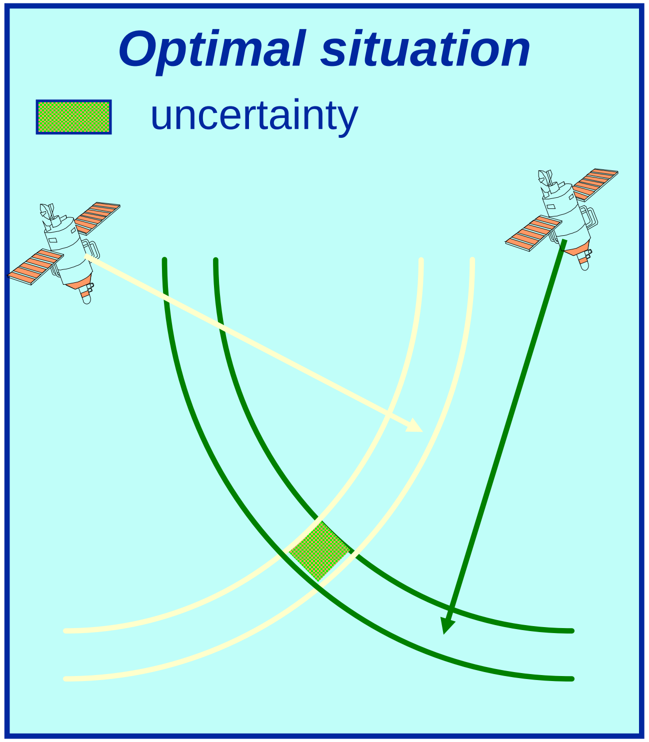
5. for **time** : TDOP = $(D_{44}/c)^{1/2} = \sigma_{cdt} / \sigma_{UERE}$

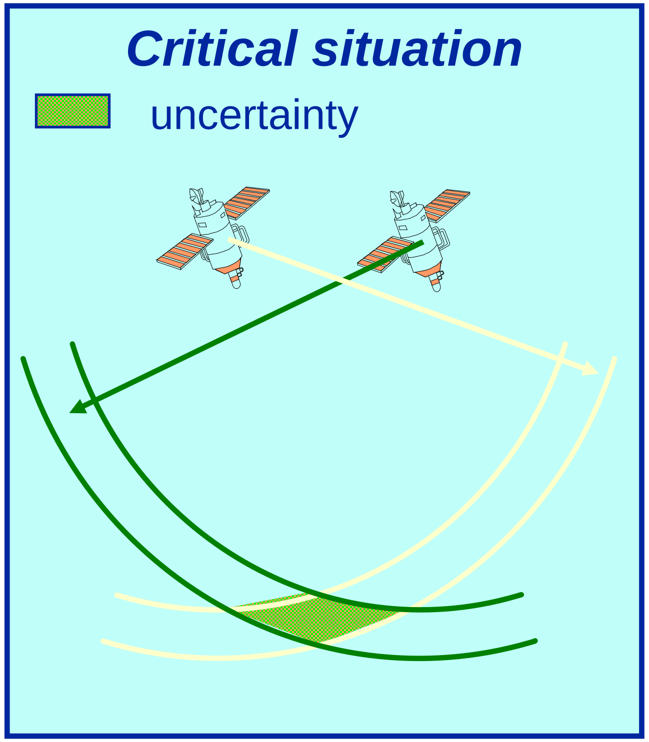
DOP dilution of precision

The relative satellites / receiver geometry has impact on the final accuracy



DOP dilution of precision

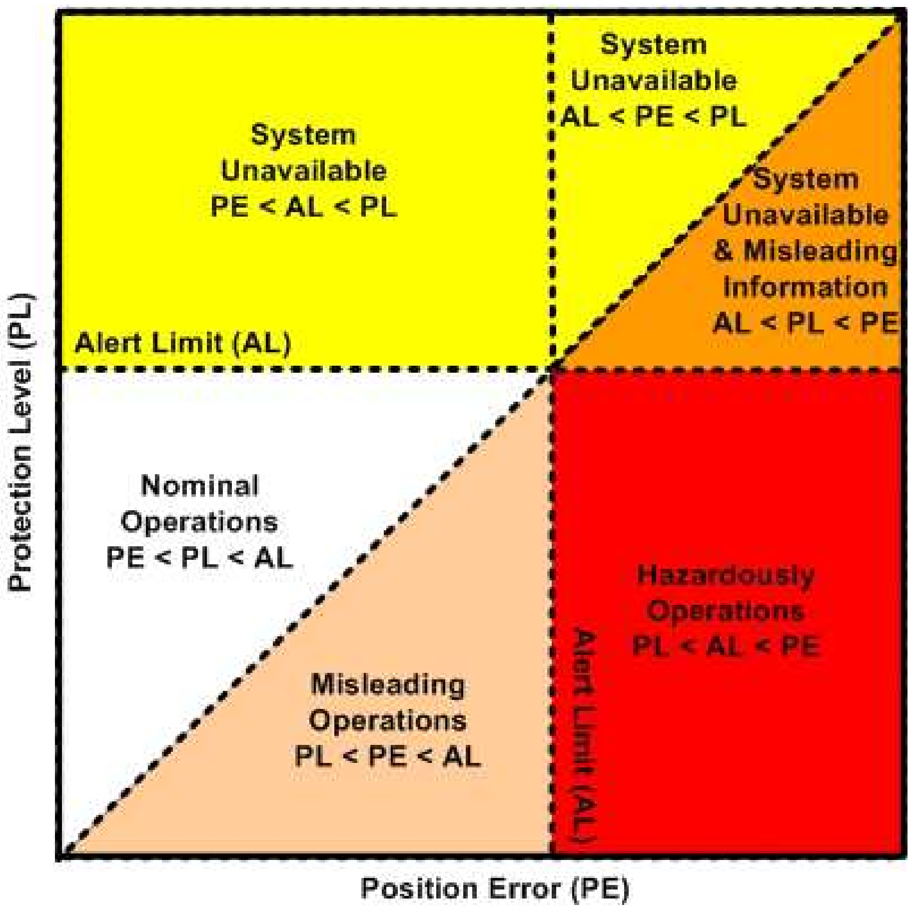




Geometrical illustration

The Stanford diagram

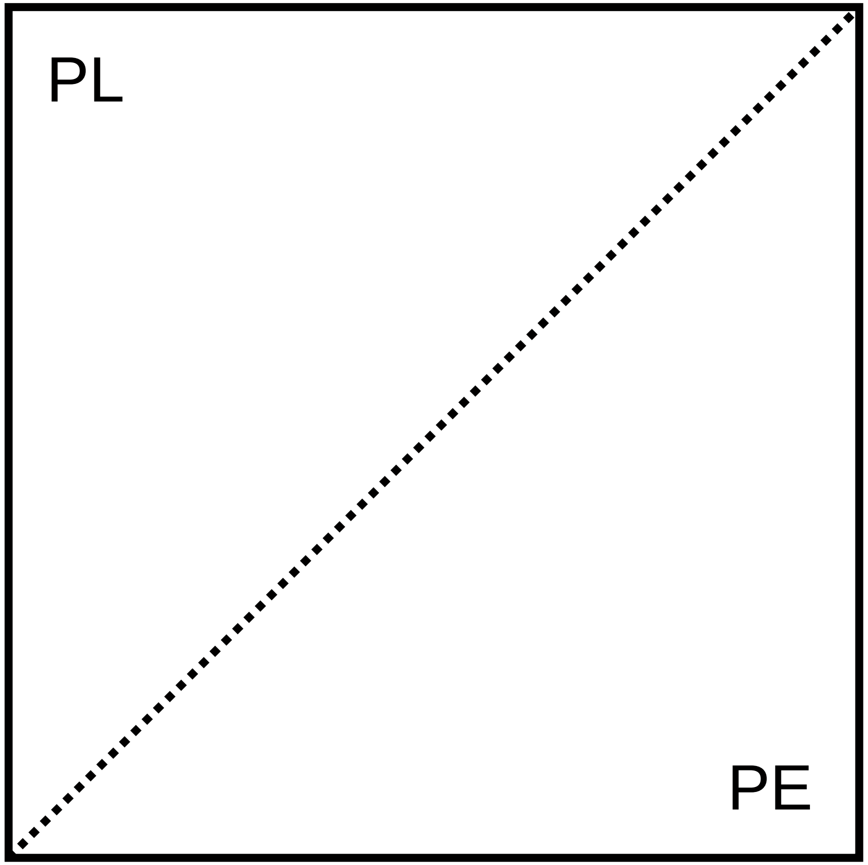
Hazardously Misleading Information (HMI) is an integrity event occurrence when, being the system declared available, the position error exceeds the Alert Limit



The Stanford diagram

- Misleading Information (MI) should not exceed in occurrence a targeted probability
- This probability is called the **integrity risk**
- While PL should keep reasonable in value (not over-bounding)

Diagonal = “magic” PL



Integrity in nominal conditions

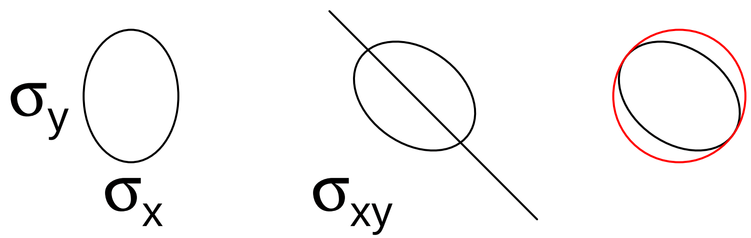
The range errors σ propagate to **x** through the LS system with:

$\mathrm{cov}(\mathbf{dx}) = (H^T H)^{-1} \sigma^2_{UERE}$ **(15)**

Let us examine the error in **norm**: it follows a *khi² distribution with 4 dimensions* and this law gives a statistical bound which contains e.g. 99.999% (10^{-3} integrity risk) of the error in norm

In **(15)**, the maximum eigenvalue of $(H^T H)^{-1}$ gives the radius of the hypersphere tangent to the hyperellipse made by $(H^T H)^{-1}$

Illustration in 2D:



Integrity: error propagation thru LS

From **(15)** after rotation ($Q = R\,(H^T\,H)^{-1}\,R^{-1}$) and the khi² table:

$$4DPL = 6.18\sqrt{\max(eig(Q))}\;\sigma_{UERE} = 6.18\,\sigma_x \qquad \textbf{(16)}$$

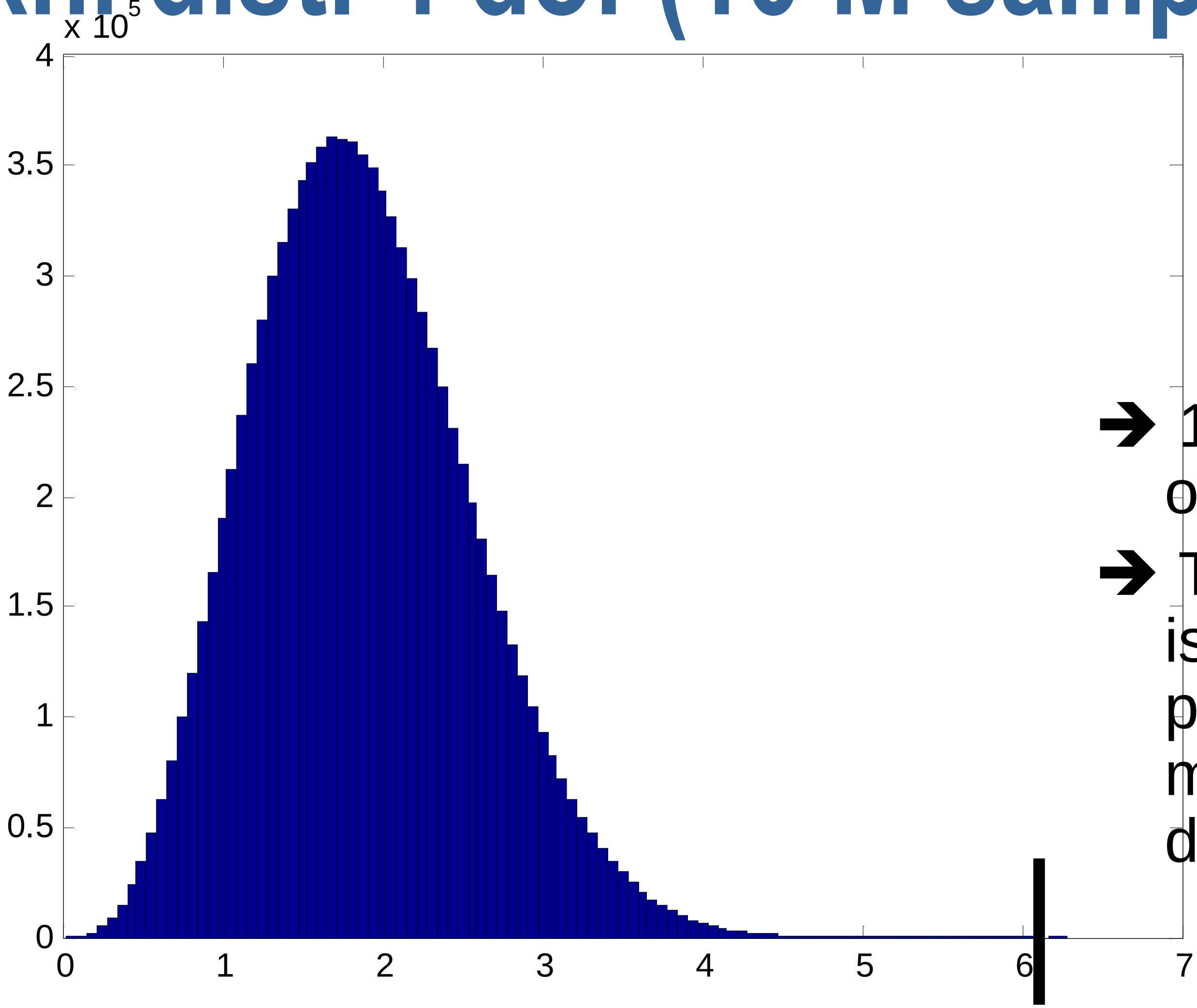
$$3DPL = 6.18\sqrt{\max(eig(Q(1{:}3,1{:}3)))}\;\sigma_{UERE} \qquad \textbf{(17)}$$

$$\mathbf{HPL = 6.18\sqrt{\max(eig(Q(1{:}2,1{:}2)))}\;\sigma_{UERE}} \qquad \textbf{(18)}$$

$(K_{PMD} = 6.18)^2$ is read in the khi² table, 4 dof, 10^{-7} integrity risk

Note that σ_{UERE} characterizes the range error *a priori*

Hist. khi dist. 4 dof (10 M samples)



- ➔ 1 sample over 10 M
- ➔ This risk is a PMD prob. of missed detection

Integrity in non-nominal conditions

Fault detection and exclusion

Form residuals and their norm (Normalized Sum of Squared Errors):

$$NSSE = (\rho - H\,\mathbf{x})^T (\rho - H\,\mathbf{x}) / \sigma^2_{UERE} \qquad \textbf{(19)}$$

The test statistic NSSE will be compared to a threshold T

T(N-4, PFA) is a function of the number of degrees of freedom i.e. satellites (N) – the number of unknown (4) and the desired probability of false alarm PFA

Classical FDE mechanism

- Fault detection if NSSE > khi² threshold T for selected PFA
- Fault exclusion:
- for **sv1**, form its normalized: $\text{residual}_1 \,/\, \sqrt{S(i,i)}$

  ...
- for **svN**, form its normalized: $\text{residual}_N \,/\, \sqrt{S(N,N)}$
- finally remove the satellite with maximum normalized residual

with $S = I - H(H^T H)^{-1} H^T$ **(20)**

and iterate (while 4 svs)…