SaPPART Training School
Fundamentals of GNSS for ITS applications

GPS observations and positioning

David Bétaille
Ifsttar

9-11 May 2016
University of Aveiro
Portugal

Contents

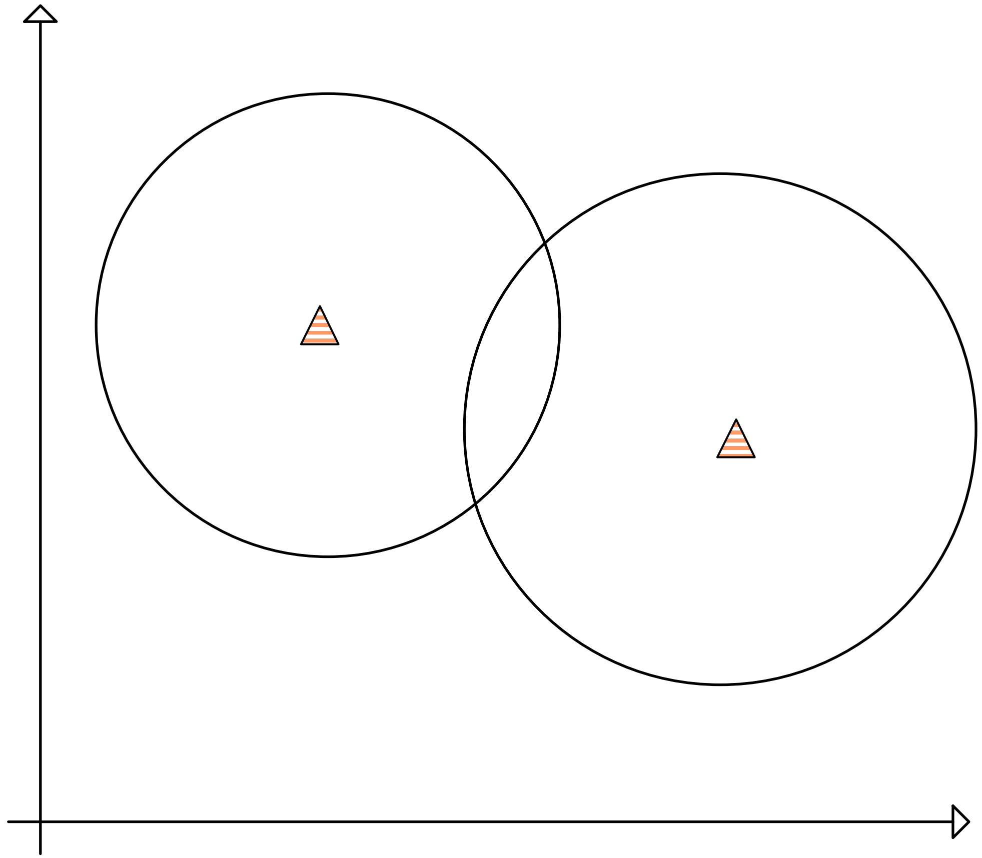
- GPS observations
- Code
- Phase
- GPS positioning
- Using code
- Standard
- Differential
- Kinematic GPS
- Using phase and solving ambiguities

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Trilateration

- measuring distances to known points (beacons)



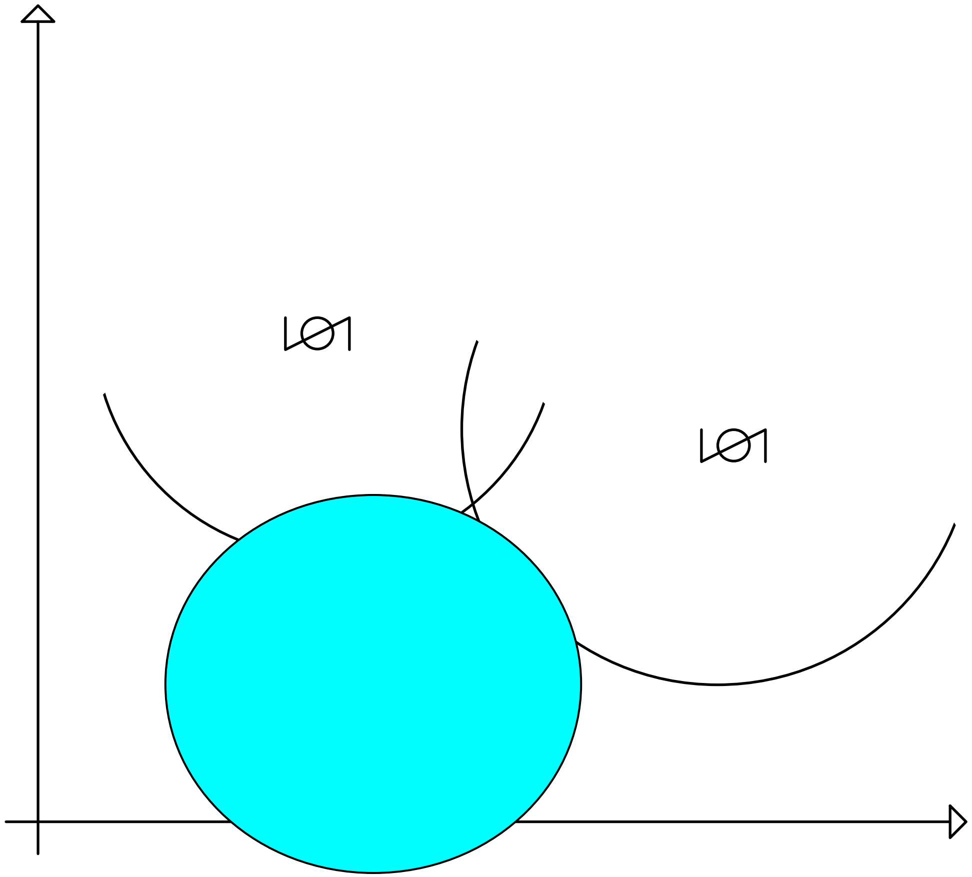
2D (∩ cercles) => 2 solutions

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Trilateration

- measuring distances to known points (beacons)



2D (∩ cercles) top-down sat to earth signal => 1 solution

COST is supported by the EU Framework Programme Horizon 2020

Trilateration

Solution of the problem

2 unknown ➔ 2 points (beacons: x_1y_1 and x_2y_2) are needed

More points give redundancy (over-determination):

$$\rho_1 = \sqrt{(x_1-x_u)^2 + (y_1-y_u)^2}$$

$$\rho_2 = \sqrt{(x_2-x_u)^2 + (y_2-y_u)^2}$$

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

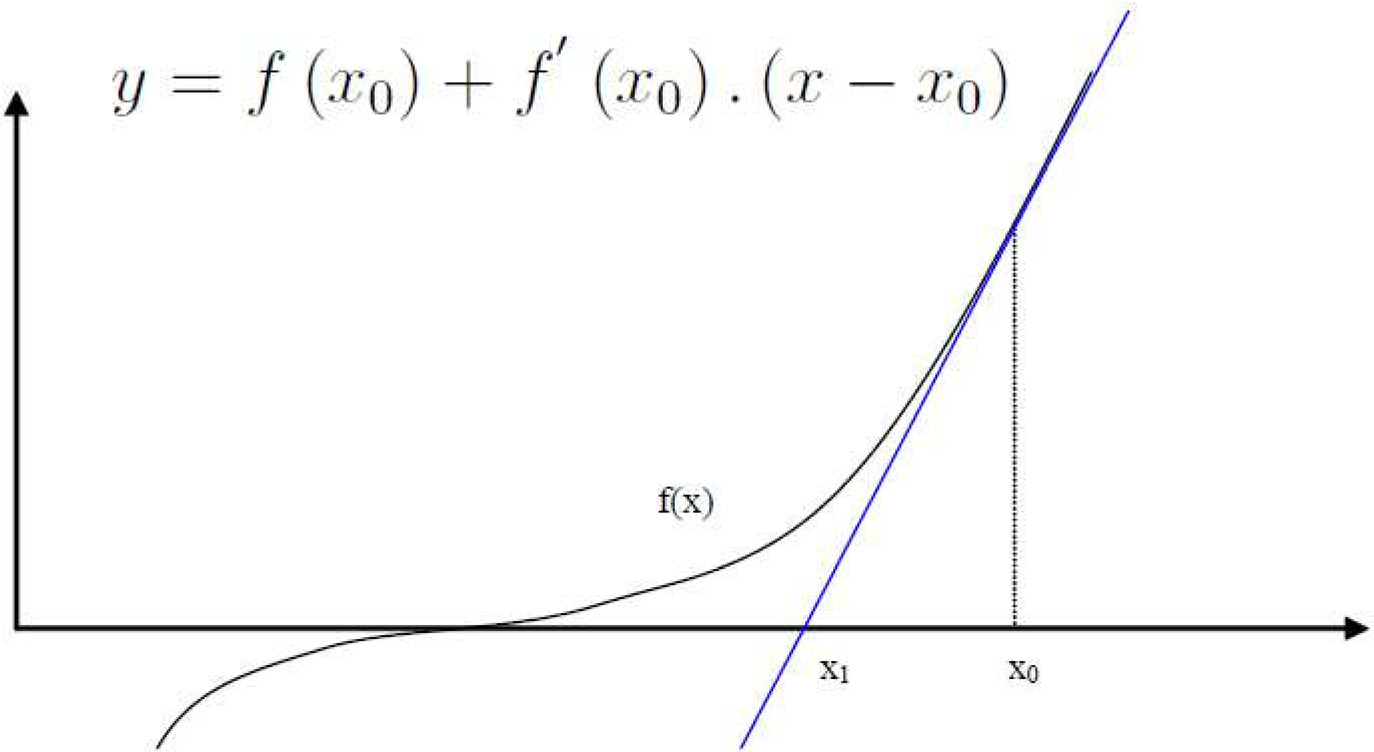
COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Iterative method (Newton-Raphson)

<u>Like solving f(x)=0 with a scalar non linear fonction f</u>

$$y = f(x_0) + f'(x_0).(x - x_0)$$



COST is supported by the
EU Framework Programme
Horizon 2020

Linearization

<u>In the vicinity of an approximate $x_o y_o$ solution</u>

Newton-Raphson derivation:

$$\rho_1 = \sqrt{(x_1-x_o)^2 + (y_1-y_o)^2} + d\rho_1/dx\,(x_u-x_o) + d\rho_1/dy\,(y_u-y_o)$$
$$\rho_2 = \sqrt{(x_2-x_o)^2 + (y_2-y_o)^2} + d\rho_2/dx\,(x_u-x_o) + d\rho_2/dy\,(y_u-y_o)$$

$$\rho_1 = \rho_{1o} - (x_1-x_o)/\rho_{1o}\,(x_u-x_o) - (y_1-x_o)/\rho_{1o}\,(y_u-y_o)$$
$$\rho_2 = \rho_{2o} - (x_2-x_o)/\rho_{2o}\,(x_u-x_o) - (y_2-x_o)/\rho_{2o}\,(y_u-y_o)$$

$$\begin{bmatrix} \rho_1 - \rho_{1o} \\ \rho_2 - \rho_{2o} \end{bmatrix} = \begin{bmatrix} -(x_1-x_o)/\rho_{1o} & -(y_1-x_o)/\rho_{1o} \\ -(x_2-x_o)/\rho_{2o} & -(y_2-x_o)/\rho_{2o} \end{bmatrix} \begin{bmatrix} x_u-x_o \\ y_u-y_o \end{bmatrix}$$

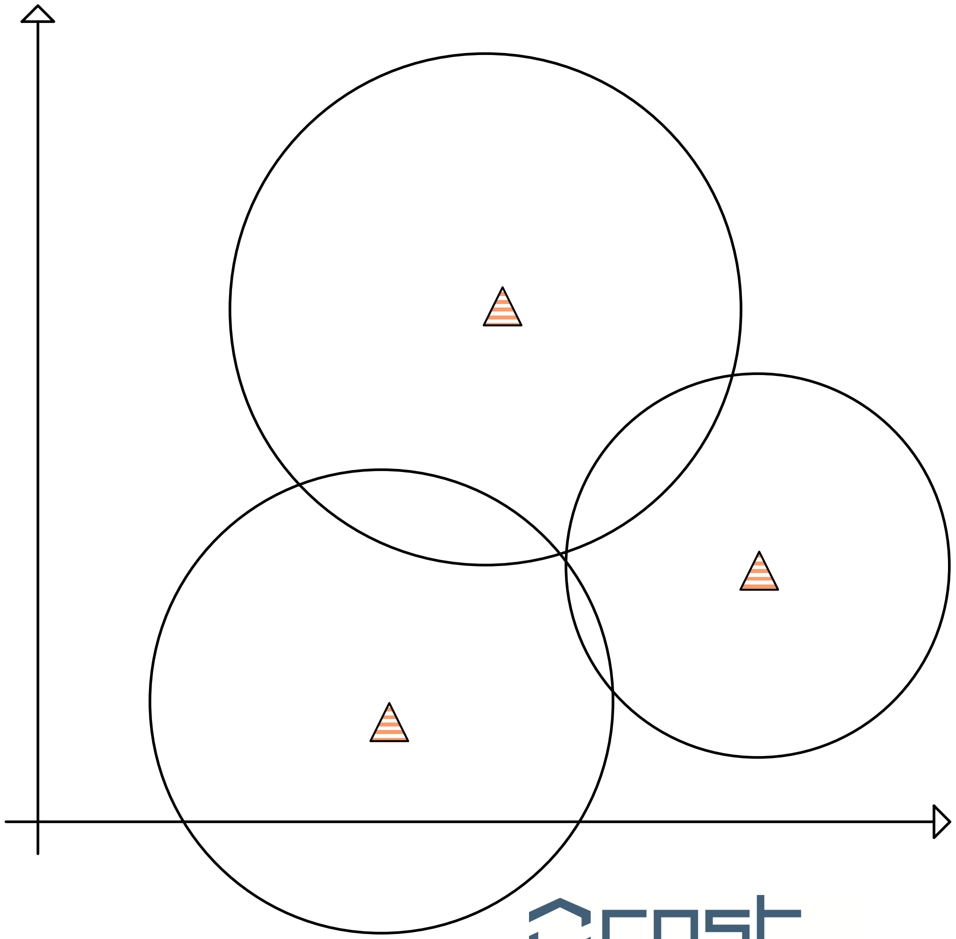
$Y = AX$

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Over-determination

- measuring distances to **more** known points than strictly needed

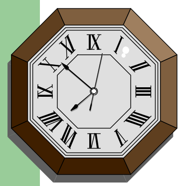


2D (∩ cercles)
redundancy
=> test meas,
optimization…

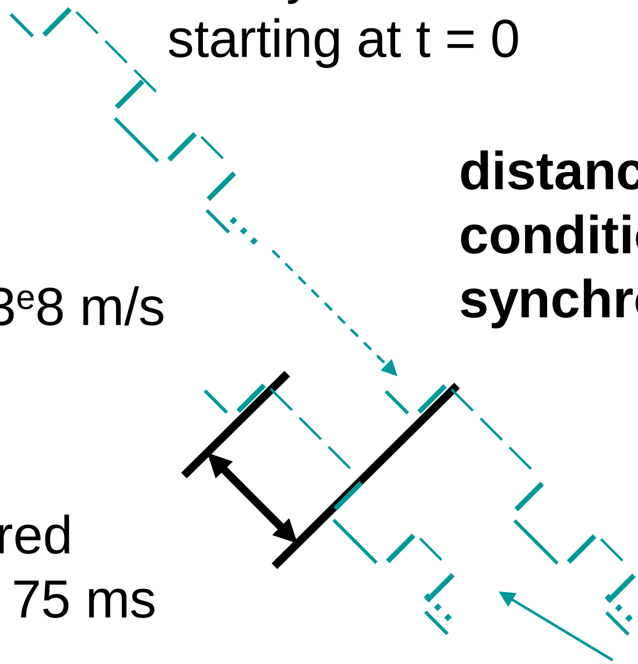
COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Measuring satellite-receiver distance



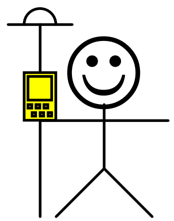
Satellite:
binary code
starting at t = 0

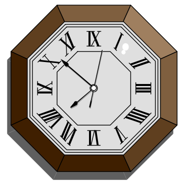


distance ~ 75e-6 x 3e8 = 22500 km conditionnally to a perfect synchronization of the clocks

speed of light ~ 3e8 m/s

Measured
delta = 75 ms
(e.g.)





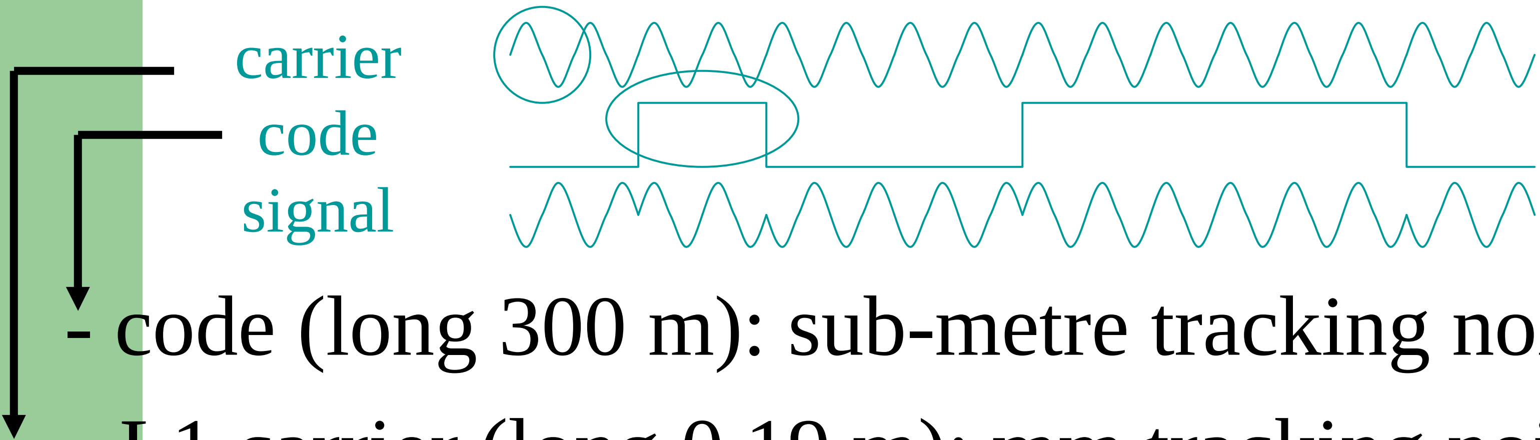
Receiver: same code

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

GPS code and phase (for L1 carrier)

GPS L1 (1.5 GHZ) decomposition:



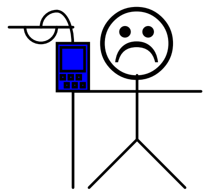
- code (long 300 m): sub-metre tracking noise level
- L1 carrier (long 0.19 m): mm tracking noise level
- travel time and distance with ambiguity (modulo 2π): one measures the variation of this distance

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Main problems…

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



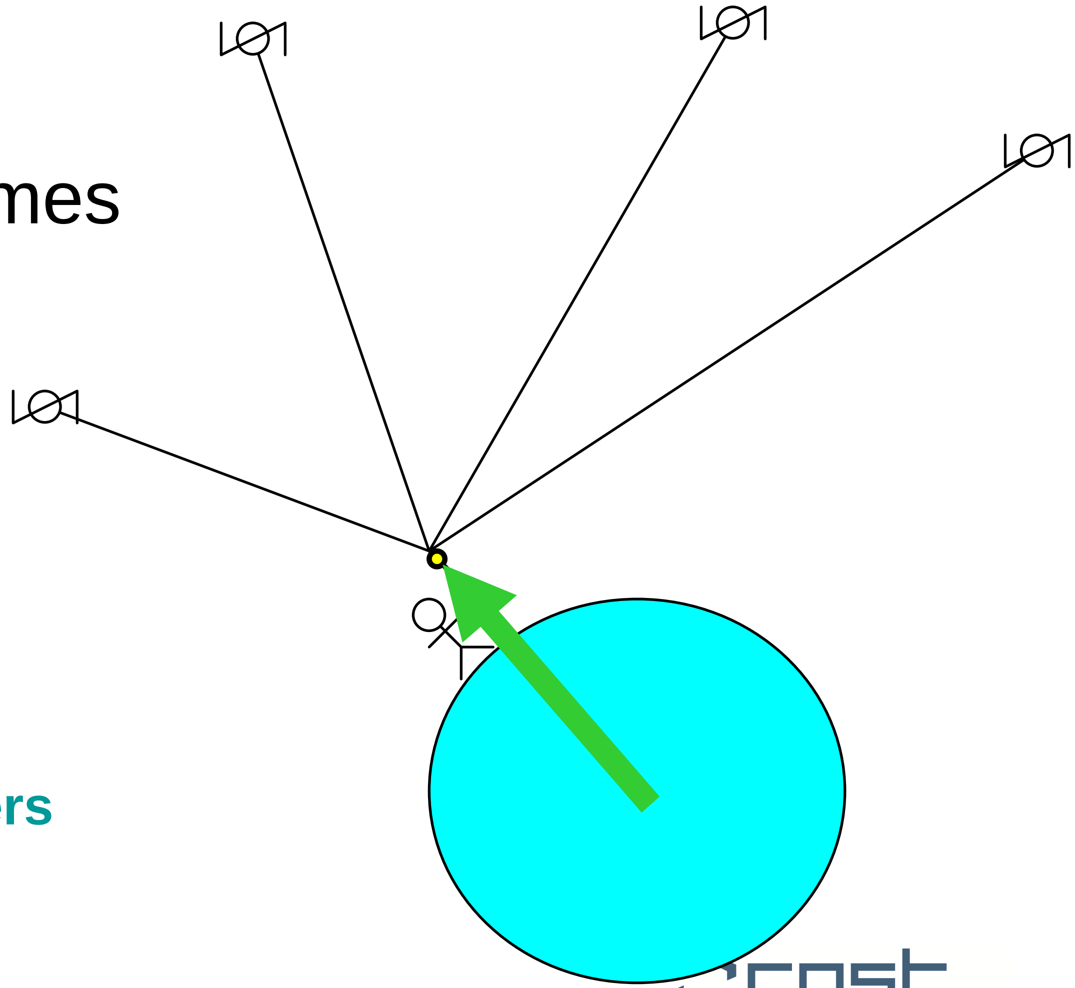
COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Standard GPS positioning service

- using code
- measuring travel times

i.e. distances



precision: **a few meters**

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

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…).

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

GPS positioning / code (2)

Observation equation and position of the problem

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

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Linearization (1)

The function f is non linear and it should be linearized

- Let **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:

$$f_j(x_u, y_u, z_u, dt) = f_j(x'_u+\Delta x_u , y'_u+\Delta y_u , z'_u+\Delta z_u , dt'+\Delta dt) \quad \textbf{(4)}$$

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Linearization (2)

- 1st order Taylor development:

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

- Derivatives of **(2)** are:

$$\frac{\partial f(x'_u, y'_u, z'_u, dt')}{\partial x'_u} = -\frac{x_j - x'_u}{r'_j}$$

$$\frac{\partial f(x'_u, y'_u, z'_u, dt')}{\partial y'_u} = -\frac{y_j - y'_u}{r'_j}$$

$$\frac{\partial f(x'_u, y'_u, z'_u, dt')}{\partial z'_u} = -\frac{z_j - z'_u}{r'_j}$$

and $$\frac{\partial f(x'_u, y'_u, z'_u, dt')}{\partial dt'} = c$$

with: $r'_j = \sqrt{(x_j - x'_u)^2 + (y_j - y'_u)^2 + (z_j - z'_u)^2}$

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Linearization (3)

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

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

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 ρ'_j is known and ρ_j is measured)

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Linearization (4)

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

$$a_{yj} = (y_j - y'_u)/r'_j$$

$$a_{zj} = (z_j - z'_u)/r'_j$$

ρ'_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:

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

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

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

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

GPS positioning / code (4)

General case (> 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 $(H^T H)^{-1}$: this is the pseudo-inverse or generalized inverse of H:

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

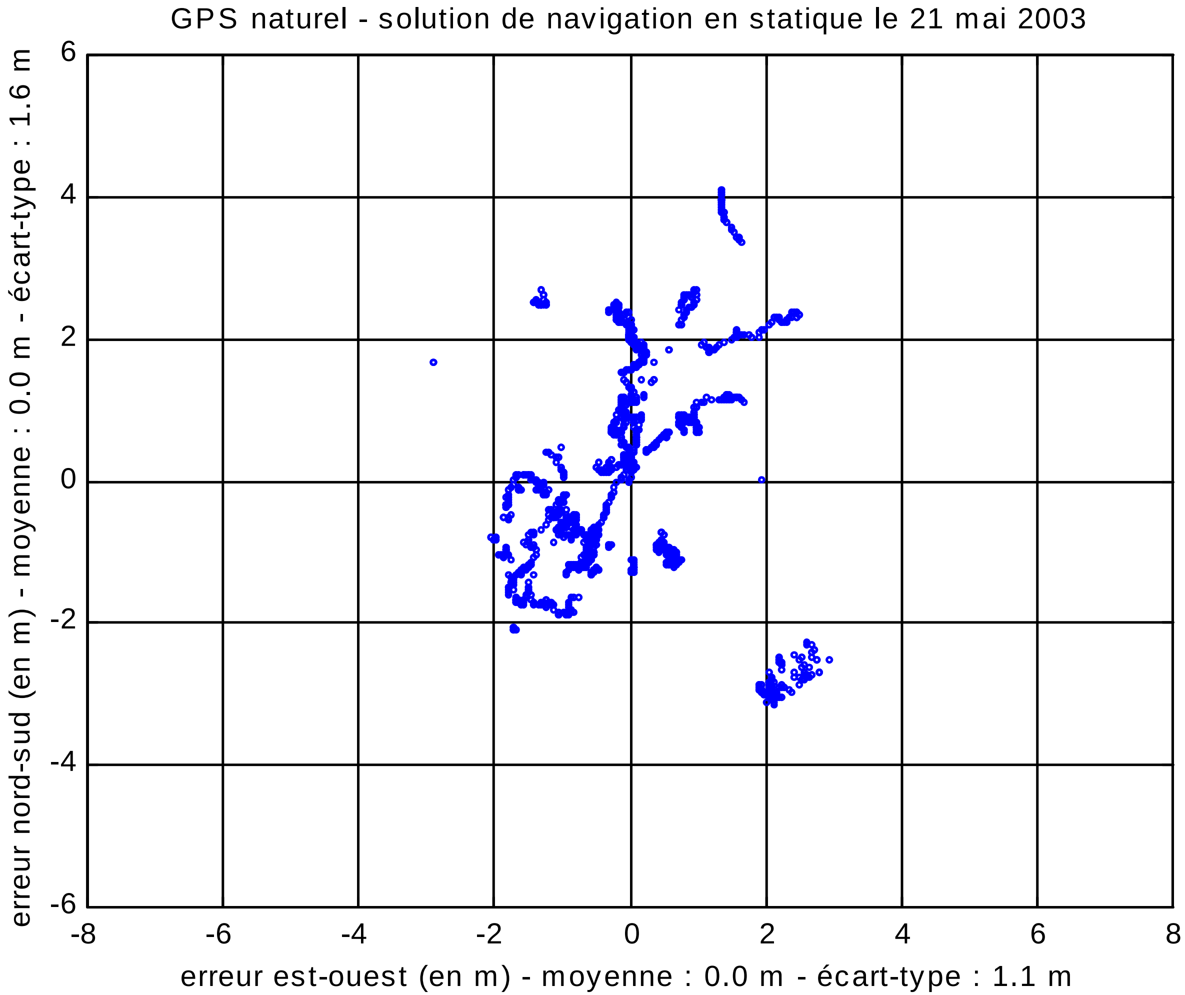
Note: one demonstrates that this solution minimizes the sum of the squared residuals (so called least-squares solution)

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Standard GPS positioning service

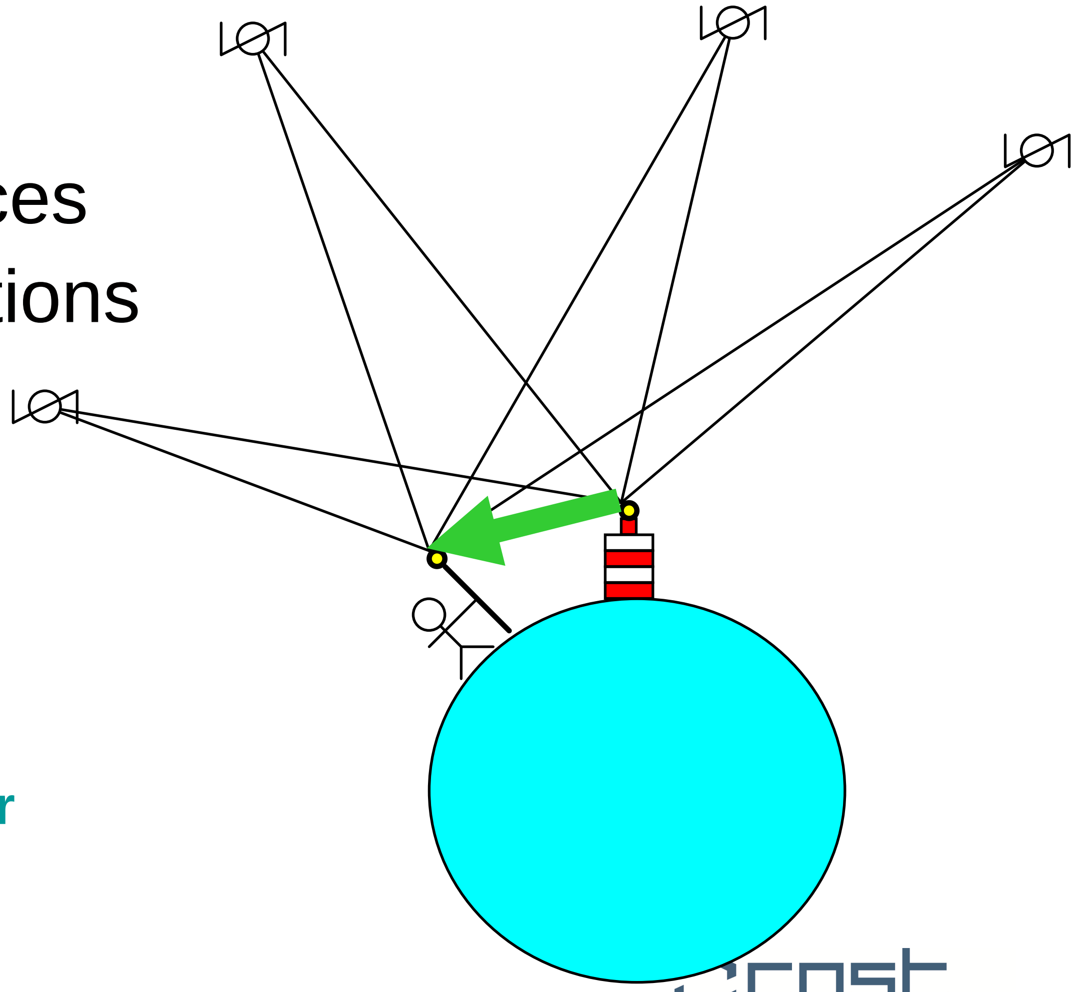
Diagram of the horizontal error typical of the GPS standard service, for 24h. Observations collected near Nantes, in may 2003



COST is supported by the
EU Framework Programm
Horizon 2020

Differential GPS

- using code
- measuring distances
- correlated perturbations (base and rover)
- using one base or a network



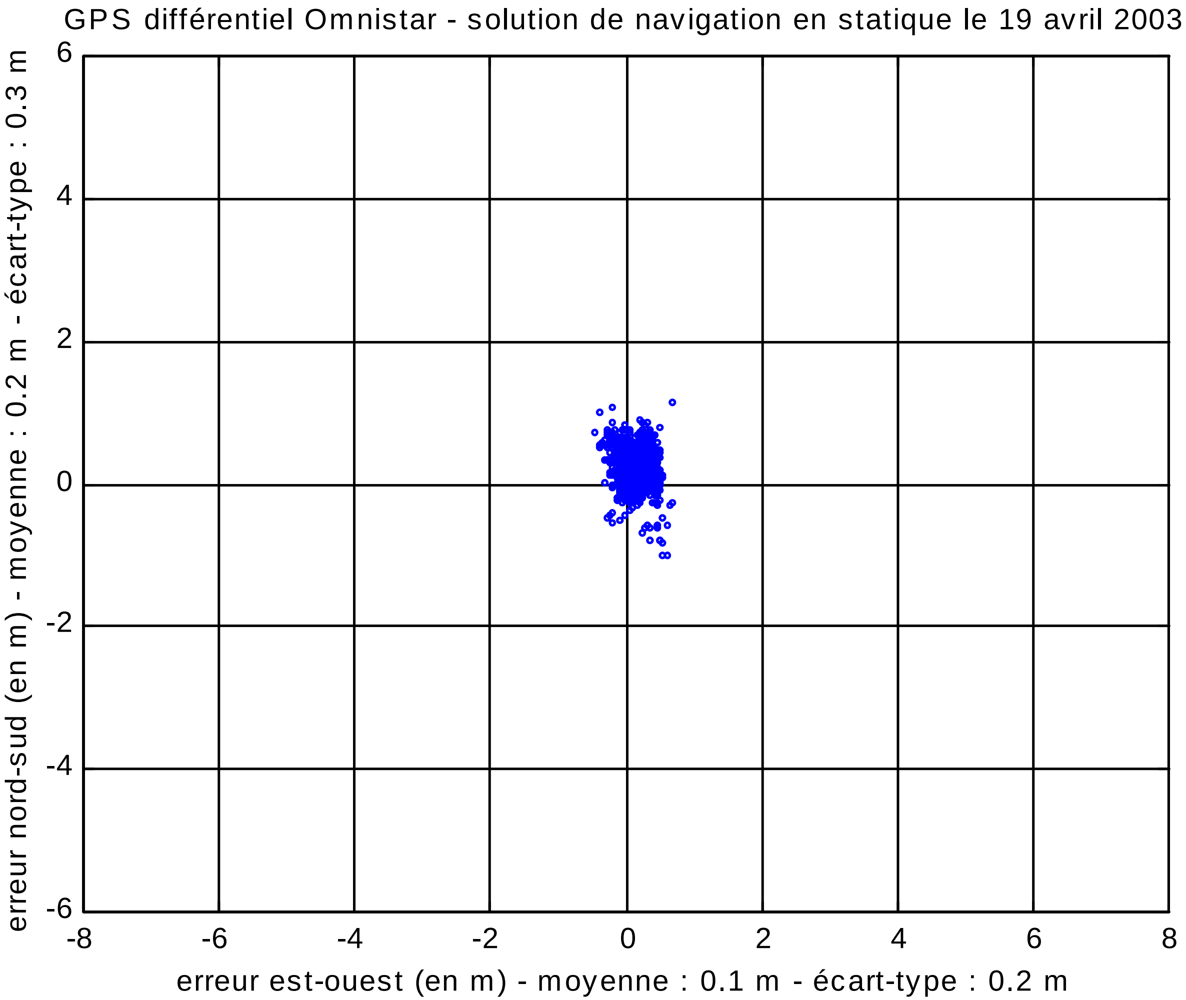
precision: (sub-)**meter**

COST is supported by the EU Framework Programme Horizon 2020

cost EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Omnistar DGPS positioning service

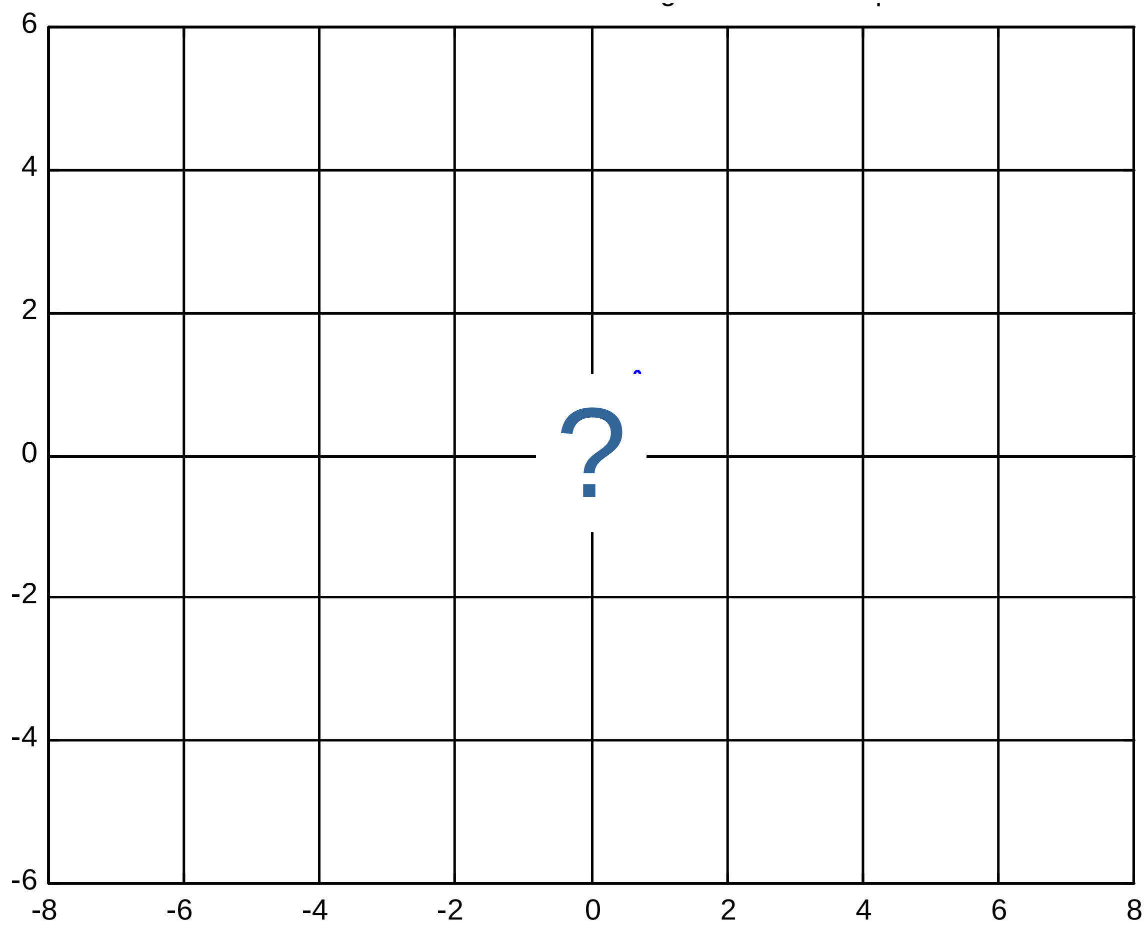
Diagram of the horizontal error typical of the DGPS Omnistar network, for 24h. Observations collected near Nantes, in april 2003



COST is supported by the
EU Framework Programm
Horizon 2020

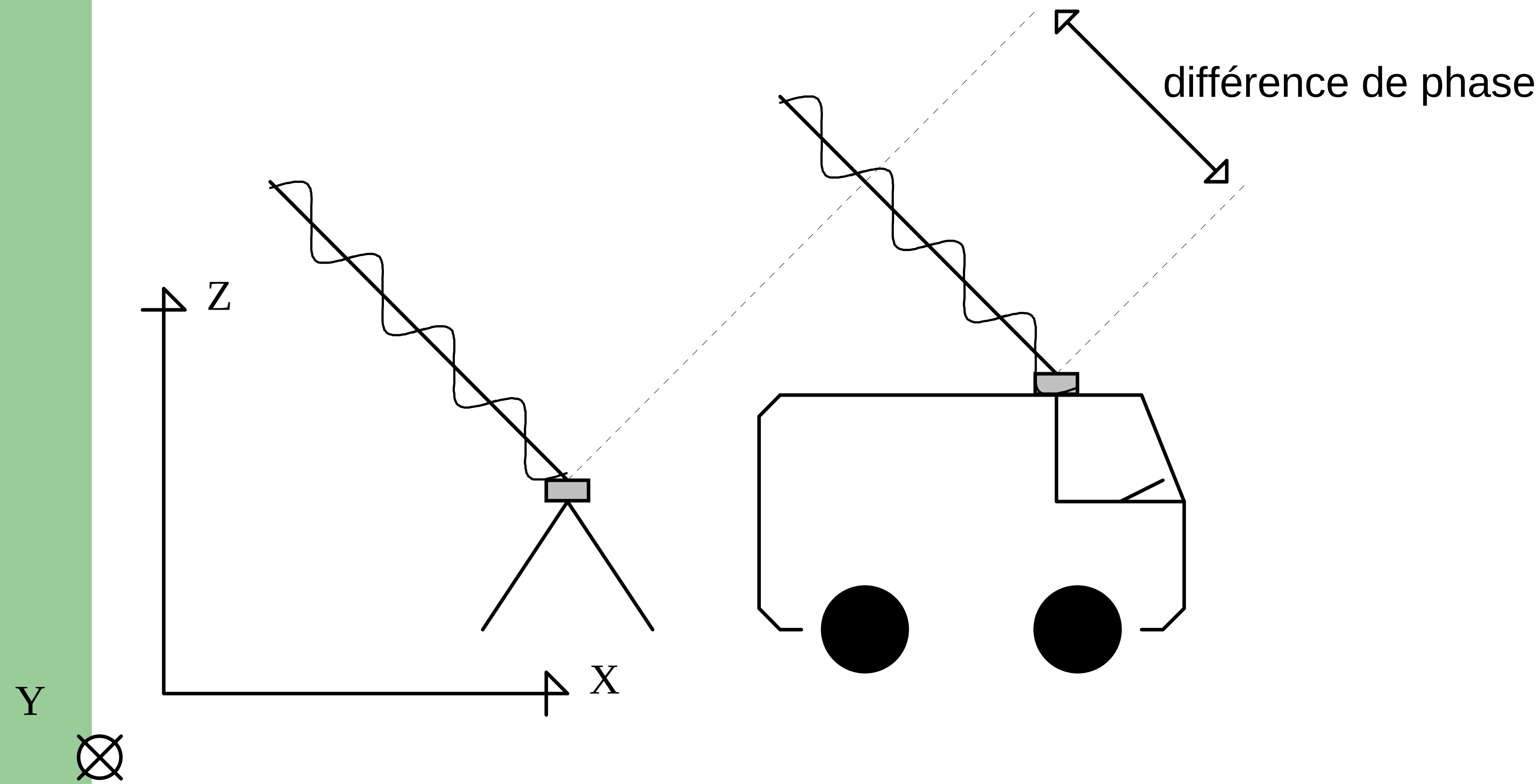
Can GPS be more accurate?

Yes!
Kinematic GPS
(in post-processing
or in real-time)
using the signal
carrier phase



COST is supported by the
EU Framework Programm
Horizon 2020

Interferometry bw base and rover



COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

GPS positioning / phase (1)

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

$\lambda\phi^j = R^j + \lambda N^j$ **(3)** (λ: wave length ~ 0.19 m)

- ϕ^j: measured carrier phase (from t0)
- N^j: integer ambiguity at t0 (invariant with time)
- R^j: true geometric distance bw the receiver and the satellite j: $R^j = ||\mathbf{s}^j - \mathbf{u}||$

Note: one has not accurate enough models of the atmospheric delays to use carrier phase ➔ so, one's own observations will be combined with other observations, synchronous, from another receiver relatively close, so that these delays (as well as satellites and receivers clock terms) be eliminated, and the other error sources are neglected (multipath, noise).

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

GPS positioning / phase (2)

Observation equation combining base and rover

$\lambda\phi^j_m = R^j_m + \lambda N^j_m$ **(jm)** (at the rover, for every satellite j)

$\lambda\phi^p_m = R^p_m + \lambda N^p_m$ **(pm)** (at the rover, for the "pivot" sat.)

$\lambda\phi^j_b = R^j_b + \lambda N^j_b$ **(jb)** (at the base, for every satellite j)

$\lambda\phi^p_b = R^p_b + \lambda N^p_b$ **(pb)** (at the base, for the "pivot" sat.)

$\lambda DD\phi^j = DDR^j + \lambda DDN^j$ **(4)** (for every satellite $j \neq p$)

- DD : combination of the observations (jm-pm)-(jb-pb)
- p : pivot (in general, the highest satellite)

This combination is named a double difference.

COST is supported by the
EU Framework Programme
Horizon 2020

GPS positioning / phase (3)

Equation in DD and position of the problem

If there are N (>1) satellites, there are (N-1) DD of unknown ambiguities, plus 3 unknown rover coordinates (the base coordinates are known), but only (N-1) equations ➔ several « epochs » are required.

A multi-epoch system has to be solved with (N-1)*nbep equations and (N-1)+3 unknown: x_u, y_u, z_u, if the rover does not change its position from one epoch to the next.

With N satellites, the problem is solved if: nbep > 1+3/(N-1).

$$\lambda DD\phi^{j\neq pivot} = f^{j\neq pivot}(x_u, y_u, z_u) + \lambda DDN^{j\neq pivot} \quad (5)$$

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

GPS positioning / phase (4)

<u>If the rover changes its position…</u>

A multi-epoch system has to be solved with (N-1)*nbep equations and (N-1)+3*nbep unknown: x_u, y_u, z_u, variable with time if the rover changes its position from one epoch to the next.

The problem is solved if: (N-1)*nbep > (N-1)+3*nbep.

From this inequality, one shows that N > 1+3*nbep/(nbep-1).

This is known as fixing ambiguities “On The Fly”: 5 satellites minimum are required.

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Linearization and fixing ambiguities

- For k epochs, next matrix equation is obtained:

$$\begin{bmatrix} \Delta\rho_{1(k1)} \\ \cdots \\ \Delta\rho_{N-1(k1)} \\ \cdots \\ \Delta\rho_{1(kep)} \\ \cdots \\ \Delta\rho_{N-1(kep)} \end{bmatrix} = \begin{bmatrix} a_{x1(k1)} & a_{y1(k1)} & a_{z1(k1)} & \lambda & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{x1(k1)} & a_{y1(k1)} & a_{z1(k1)} & 0 & 0 & \lambda \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{x1(kep)} & a_{y1(kep)} & a_{z1(kep)} & \lambda & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{x1(kep)} & a_{y1(kep)} & a_{z1(kep)} & 0 & 0 & \lambda \end{bmatrix} \begin{bmatrix} \Delta xu \\ \Delta yu \\ \Delta zu \\ DDN_1 \\ \cdots \\ DDN_{N-1} \end{bmatrix}$$

$\Delta\rho$ = H [Δxu, Δyu, Δzu DDN_1 … DDN_{N-1}]' = A Δx + G n (6)

whose 2nd bloc will be solved in real then in integer spaces

COST is supported by the
EU Framework Programme
Horizon 2020

EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Linearization and fixing ambiguities

- Complex problem (linearized equations under constraints)

- Float solution (dm accurate) while solving both the position and the ambiguities in real space

- Fix solution (cm accurate) with the ambiguities constrained in integer space

- Once the ambiguities fixed, the system (6) is solved iteratively until convergence: the final accuracy gets sub-cm with fixed ambiguities, and this works until distances bw base and rover of tens of km in double difference

$$\Delta\rho - G\,n = A\,[\Delta x_u, \Delta y_u, \Delta z_u] \quad (7)$$

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY

Further reading…

- **“GPS localisation et navigation par satellites”, F. Duquenne, S. Botton, F. Peyret, D. Bétaille, P. Willis, édition Hermès Lavoisier, 2005.**
- “GPS for geodesy”, by P.J.G. Teunissen and A. Kleusberg, published by Springer, 1996.
- Web sites of: Trimble, Leica, Novatel, Thales, Javad, etc…
- Revue GPS world
- Proceedings of IEEE ION conferences

COST is supported by the EU Framework Programme Horizon 2020

EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY