Hoy: última clase de teoría.

Nos quedan otras 2 clases más de este tema

- El próximo viernes **(11 oct)** tendremos el último LAB trabajando en lo que será la entrega final: usar datos de una estación durante varias horas, cálculo de sus soluciones, cambio de coordenadas, visualización de errores, etc.

Hay que llegar teniendo algún tipo de get_pos() funcionando.

- El siguiente viernes **(18 oct)** tendremos la prueba individual de esta parte:
  - Un pequeño test con respuestas tipo SI/NO y algún problema de respuesta corta (10-15 min sin apuntes). 20%
  - Un ejercicio computacional (similar a los que hemos hecho en clase) donde podéis disponer de todo el material. 80%

TENED OPERATIVAS VUESTRAS FUNCIONES

- Para la entrega final tendréis una semana adicional.

Objetivo 1ª Clase de Lab (TAREA 1)

<u>Escribir función [XYZ, cdT]=interp_sat(sp,t,PRN,N)</u>

Parámetros de entrada: **sp = estructura de datos de fichero SP3**

t = TOW (cae dentro de datos de sp)

PRN = ID de un satélite.

N = número de nodos a usar.

SALIDA:

- Vector 3 x 1 con posición XYZ + Error de reloj cdT (en metros) para el satélite PRN en el tiempo t, interpolando los datos de sp usando N nodos.

Si no os funciona disponéis de mi versión (mi_fun.p) para comparar resultados y corregir vuestros errores, pero debéis tener vuestra versión operativa para la entrega final.

Objetivo 2ª Clase de Lab (TAREA 2)

Escribir función X=get_pos(sp,Tr, PRNs, pseudorangos, X, N)

ENTRADA:

- estructura sp con datos de las órbitas

- datos suministrados por el GPS:

Tiempo recepción (Tr)
Lista de satélites observados (PRNs)
Medidas tomadas(pseudorangos) a los satélites anteriores.

- Opcionalmente

X: vector 4x1 con hipótesis inicial (posición xyz + cdt).

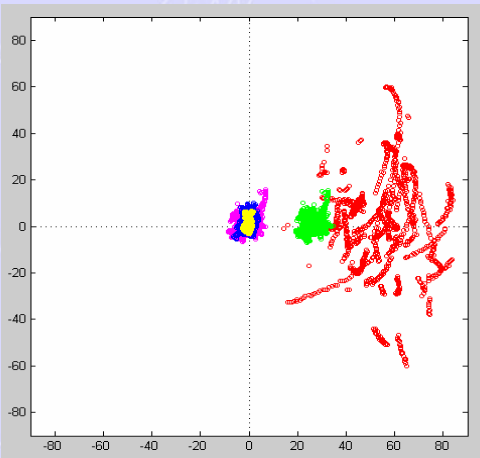
Número de nodos N a usar en la interpolación.

SALIDA: vector columna 4x1 con la solución final X = xyz + cdt

¿Qué nos queda para este último LAB?

- Comprobar que get_pos funciona y en su caso completar el algoritmo con las correcciones adicionales.

- Ir preparando la entrega final: aplicar vuestro algoritmo a una colección de observaciones para estudiar los errores resultantes.



- Necesitamos datos cuya solución conozcamos para comparar.
- Afortunadamente hay muchas estaciones (con posición conocida) publicando sus observaciones (formato RINEX).
- He leído varios de esos ficheros y he metido sus datos en estructuras de MATLAB (muy similares a la estructuras sp de los datos orbitales)

Datos de una estación

Estructura MATLAB con los siguientes campos:

- .week → semana GPS
- .tow → vector con los NT tiempos de todas las observaciones
- .prn → vector con los NSAT satélites para los cuales hay datos.
- .obs → Matriz NSAT x NT con los pseudorangos medidos por el GPS.
- .XYZ → vector 3x1 con la posición conocida de la antena → nuestra referencia

- Los datos (campo obs) son las medidas de los pseudorangos a los satélites visibles durante varias horas desde una única posición.
- Al resolver deberíamos obtener siempre la misma posición, pero los inevitables errores harán que los resultados se dispersen.
- Disponemos de la verdadera posición de la estación (campo .XYZ) para estimar el nivel de nuestros errores.

No confundir estos datos (observaciones locales de una estación) con los extraídos de los ficheros SP3 (datos orbitales globales).

estacion.tow (tiempos de las observaciones)

estacion.prn (satélites observados durante el día)	t0	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10	t11	t12	t13	t14	t15	t16	...
1	d	d	d	d	d	d	d	d	d	-	-	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-1	-	-	-	-	-	-	-	-	-	-	-
4	d	d	d	d	-	-	-1	-	d	d	d	d	d	d	d	d	d	d
5	-	-	d	d	d	d	d	d	d	d	-	-	-	-	-	-	-	-
7	-	-	-	-	-	-	-1	-	-	d	d	d	d	d	d	d	d	d
8	-	-	-	-	-	d	d	d	d	d	d	d	d	-	-	-	-	-
…	d	d	d	d	d	-	-1	-	-	-	-	-	-	-	-	-	-	-
	-	-	-	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d
	-	-	-	-	-	-	-1	-	-	-	-	-	-	-	-	-	-	-
		d	d	d	d	d	d	d	d	d	d	d	-	-	-	-	-	-

estacion.obs

Observaciones en un mismo tiempo (t1)

Observación = 20134456.3

No visibles: observación = -1

Procesar datos de una estación

Reservo matriz S de tamaño: 4 (X,Y,Z,cdt) x NT para guardar soluciones

Barrer desde k=1:NT (todos los instantes de tiempo disponibles):

1) Tiempo de la observación (según el receptor): Tr=estacion.tow(k)
2) Extraer del campo .obs la k-ésima columna de medidas:

   medidas = estacion.obs(:,k)

3) Determinar las observaciones validas (pseudorrangos positivos):

   ok = (medidas >0)

4) Extraer los pseudorrangos validos: obs =medidas(ok);
5) Extraer los PRN de dichos satélites: sats=estacion.prn(ok)
6) Pasar estos datos (Tr, sats, obs) como argumentos a get_pos().
7) Guardar la solución (4x1) que nos devuelve get_pos() en la k-ésima columna de la matriz S de resultados.

Procesar datos de una estación

El resultado final es una matriz 4 x NT con la posición (+ cdt) obtenida para cada instante de tiempo en el que teníamos observaciones

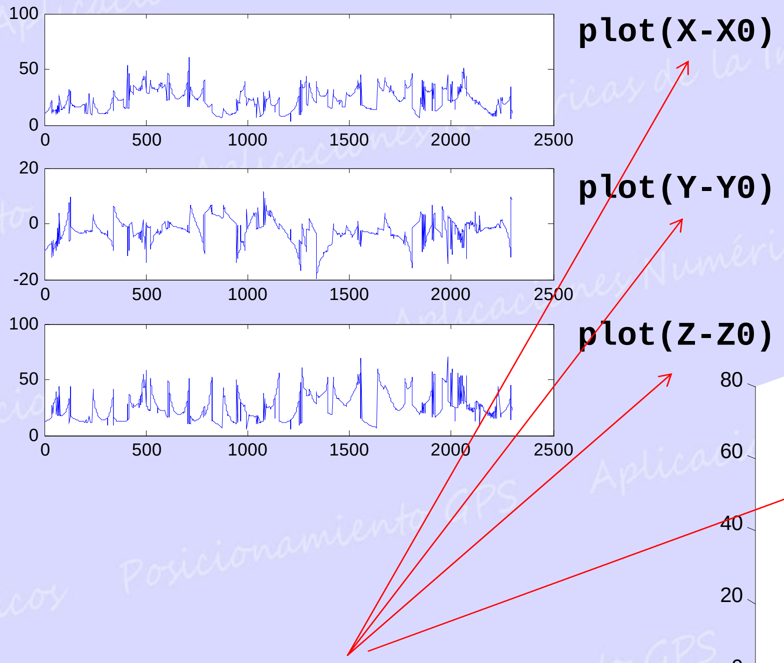
x									
y									
z									
cdt									

Al terminar el proceso tenemos NT soluciones (ligeramente distintas).

Podemos compararlas con el campo estacion.XYZ (solución correcta) para obtener los errores cometidos para analizarlos, pintarlos, etc.

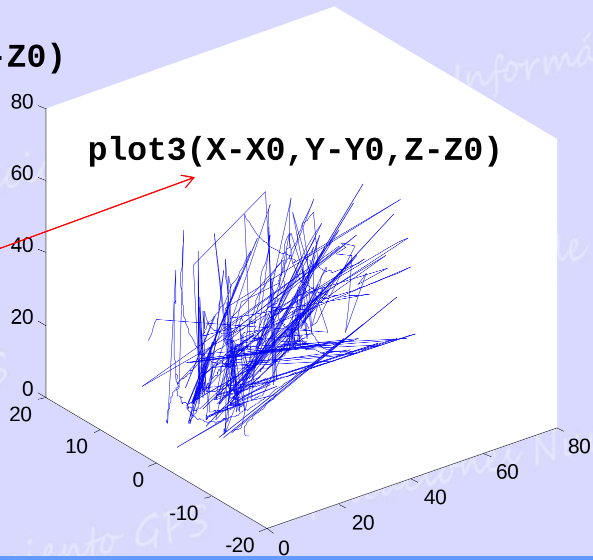
Incluso con la implementación más básica los errores no deben exceder unos 50-100 mt.

Gráficas de errores (en XYZ)



Posición conocida:

```
estacion.XYZ = [X0
                Y0
                Z0]
```



Pintar siempre las **diferencias** con la solución conocida de la estación.

Conversión de Coordenadas

Al resolver ecuaciones trabajamos con coordenadas XYZ (referidas al centro de la Tierra) pero nadie usa esas coordenadas para expresar su posición.

Hay que convertir dichas coordenadas a coordenadas angulares sobre la superficie (latitud, longitud) + altura (h) sobre la superficie.

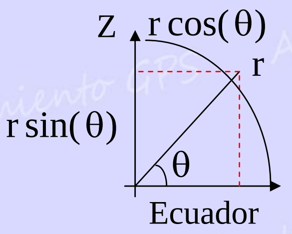
Matemáticamente: pasar de coordenadas rectangulares (xyz) → polares (θ,λ,r)

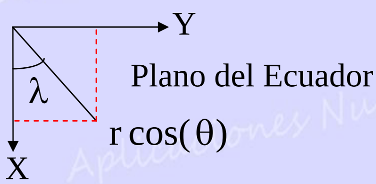
Latitud θ : ángulo con respecto al Ecuador (rango -90º a 90º).

Longitud λ : ángulo con respecto al meridiano de referencia (-180º a 180º).

Radio r : distancia al centro de la Tierra. Si le restamos el radio de la Tierra R, tenemos la altura h sobre la superficie (r=R+h)

$$X = (R+h)\cos(\theta)\cos(\lambda)$$
$$Y = (R+h)\cos(\theta)\sin(\lambda)$$
$$Z = (R+h)\sin(\theta)$$



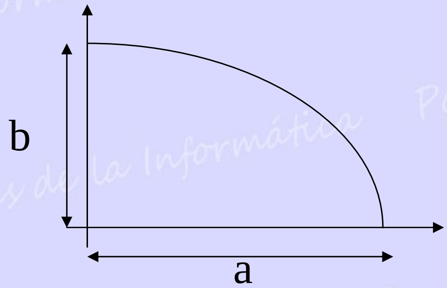


La Tierra como un Elipsoide

En realidad, la Tierra no es una esfera perfecta, es más parecida a un elipsoide

Al contrario que la esfera, cuyo único parámetro es su radio R, un elipsoide se define con dos parámetros, radio mayor (a) y radio menor (b).

La Tierra es más ancha en el Ecuador (a>b)



Alternativamente en vez de los semiejes a y b podemos usar:

Radio mayor (a) + achatamiento (flatenning, f): $f = \dfrac{a-b}{a}$

Radio mayor (a) + excentricidad e $e = \sqrt{\dfrac{a^2-b^2}{a^2}} = \sqrt{(2-f)\cdot f}$

GPS usa los datos del elipsoide WGS84 (World Geodetic Survey, 1984):

a = 6378137.000 mt b=6356752.314 mt → 1/f = 298.257224

Cambio coordenadas en un elipsoide (a, b, e)

De (θ, λ ,h) a (X, Y, Z): $N = \dfrac{a}{\sqrt{1-e^2\sin^2(\theta)}}$

$$\begin{cases} X = (N+h)\cdot\cos(\theta)\cdot\cos(\lambda) \\ Y = (N+h)\cdot\cos(\theta)\cdot\sin(\lambda) \\ Z = \left(\dfrac{N}{1-e^2}+h\right)\cdot\sin(\theta) \end{cases}$$

La transformación inversa

(X, Y, Z) a (θ, λ ,h)

no es exacta y requiere de un proceso iterativo:

$$\lambda = \arctan\frac{Y}{X}$$

Start with $h_0 = 0$

$$\varphi_0 = \arctan\frac{Z}{p\,(1-e^2)}$$

Iterate φ and h

$$N_i = \frac{a}{\sqrt{1-e^2\sin^2\varphi_i}}$$

$$h_{i+1} = \frac{p}{\cos\varphi_i} - N_i$$

$$\varphi_{i+1} = \arctan\frac{Z}{p\left(1-e^2\frac{N_i}{N_i+h_{i+1}}\right)}$$

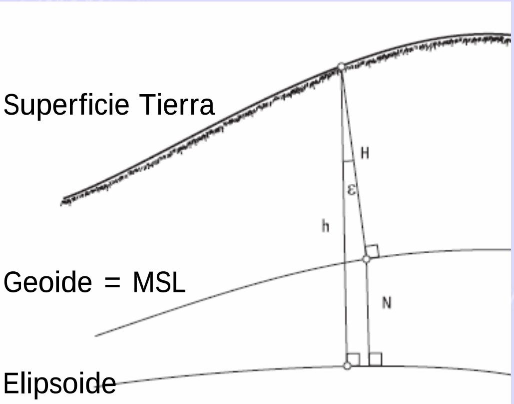
El problema adicional de la altura

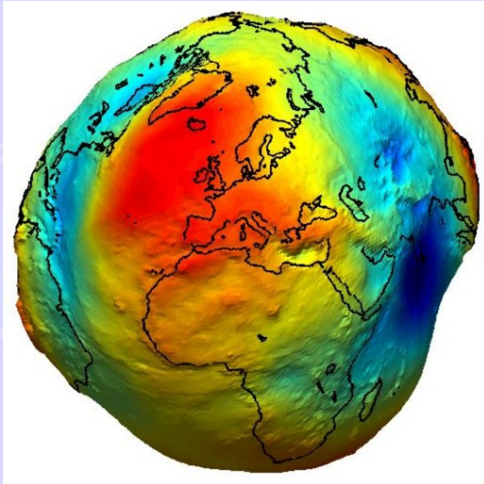
La altura h calculada es la altura sobre el elipsoide (una superficie matemática).

Nos interesa H, altura sobre el nivel medio del mar (MSL, Mean Sea Level).

El nivel medio del mar se define con el GEOIDE, una superficie irregular que recoge influencias locales de la gravedad (montañas, concentraciones de masa)

El geoide puede tener variaciones con el elipsoide WGS84 de +/- 100 metros.





Conversión (X,Y,Z) → (λ, θ, h)

Para aplicar las fórmulas anteriores os daré una función **xyz2llh** que convierte una matriz de 3xN coordenadas (X, Y, Z) en una matriz 3xN conteniendo la longitud λ (1ª fila), latitud θ (2ª fila) y altura h (3ª fila) de los N puntos dados.

El principal problema práctico es que diferentes países han usado diferentes elipsoides (y ligeramente movidos) para su cartografía (diferentes *datums*).

Estas diferencias suponen que una misma posición XYZ se transforme en latitudes/longitudes/alturas ligeramente distintas.

Además, cada país suele definir la altura respecto a un cierto nivel del mar que no es el global del geoide (en España se usa el nivel del mar en Alicante).

Esto será un problema si mezclamos datos de diferentes fuentes o al intentar representar vuestros resultados sobre un mapa (usando un datum particular).

En este curso NO NECESITAMOS preocuparnos por todo esto.

Elipsoides para todos

Name	a	b	1/f
Airy	6377563.396	6356256.909	299.324965
Airy (Modified)	6377340.189	6356034.448	299.324965
Australian National	6378160.000	6356774.719	298.250000
Bessel 1841	6377397.155	6356078.963	299.152813
Bessel 1841 (Namibia)	6377483.865	6356165.383	299.152813
Clarke 1866	6378206.400	6356583.800	294.978698
Clarke 1880	6378249.145	6356514.870	293.465000
Everest (Sabah & Sarawak)	6377298.556	6356097.550	300.801700
Everest 1830	6377276.345	6356075.413	300.801700
Everest 1948	6377304.063	6356103.039	300.801700
Everest 1956	6377301.243	6356100.228	300.801700
Everest 1969	6377295.664	6356094.668	300.801700
Fischer 1960	6378166.000	6356784.284	298.300000
Fischer 1960 (Modified)	6378155.000	6356773.320	298.300000
Fischer 1968	6378150.000	6356768.337	298.300000
GRS 1980	6378137.000	6356752.314	298.257222
Helmert 1906	6378200.000	6356818.170	298.300000
Hough	6378270.000	6356794.343	297.000000
International	6378388.000	6356911.946	297.000000
Krassovsky	6378245.000	6356863.019	298.300000
SGS 85	6378136.000	6356751.302	298.257000
South American 1969	6378160.000	6356774.719	298.250000
WGS 60	6378165.000	6356783.287	298.300000
WGS 66	6378145.000	6356759.769	298.250000
WGS 72	6378135.000	6356750.520	298.260000
WGS 84	6378137.000	6356752.314	298.257224

USA

España

Rusia

GPS

Y encima los podemos mover

Nombre datum	Elipsoide usado	dX	dY	dZ
European 1950 - Cyprus	International	-104	-101	-140
European 1950 - Eastern Regional Mean	International	-87	-96	-120
European 1950 - Egypt	International	-130	-117	-151
European 1950 - Finland, Norway	International	-87	-95	-120
European 1950 - Greece	International	-84	-95	-130
European 1950 - Iran	International	-117	-132	-164
European 1950 - Italy (Sardinia)	International	-97	-103	-120
European 1950 - Italy (Sicily)	International	-97	-88	-135
European 1950 - Malta	International	-107	-88	-149
European 1950 - Northern Regional Mean	International	-86	-96	-120
European 1950 - Portugal, Spain	International	-84	-107	-120
European 1950 - Southern Regional Mean	International	-103	-106	-141
European 1950 - Tunisia	International	-112	-77	-145

De Latitud / Longitud a Coordenadas Planas

Las coordenadas esféricas de superficie (latitud y longitud) se dan en grados con el problema de que no son unidades intuitivas ni fáciles de comparar.

Los grados de longitud son más pequeños según nos acercamos a los polos. Ni siquiera los grados de latitud (con mismo “tamaño” sobre una esfera) son iguales (expresados en km sobre la superficie) si consideramos un elipsoide.

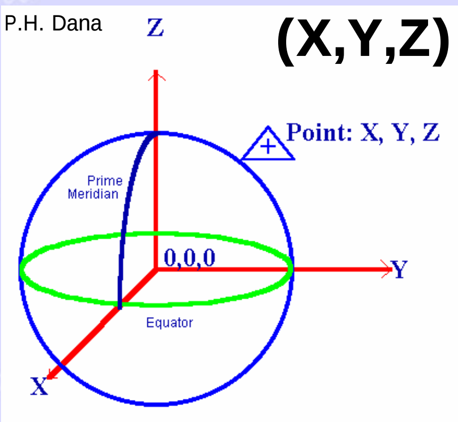
En los mapas terrestres se usan **coordenadas planas o rectangulares** (northing, easting) que se obtienen proyectando las coordenadas esféricas latitud y longitud sobre un plano.

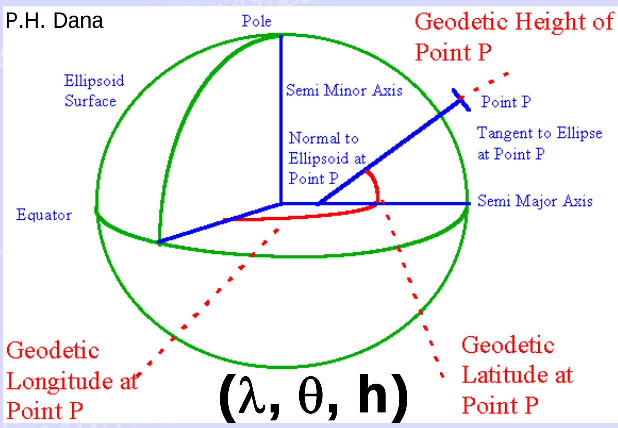
En los mapas de España usamos la proyección UTM (Universal Transversal Mercator), aunque hay muchos otros sistemas de proyección.

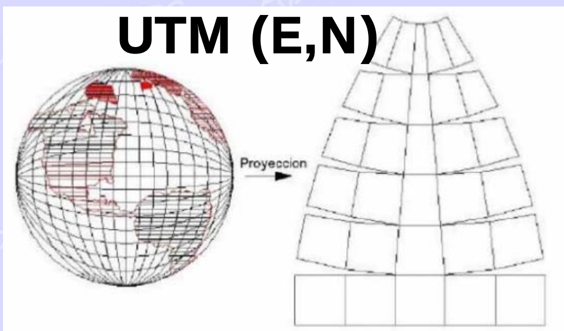
Las coordenadas planas vienen dadas en metros y son mucho más cómodas para estudiar la dispersión de las medidas obtenidas con nuestro método.

La función **ll2utm** recibe una matriz 2xN (longitud/latitud) y devuelve una matriz 2xN con las coordenadas planas UTM: este (1ª fila) y norte (2ª fila). **Ambas coordenadas UTM vienen expresadas en metros.**

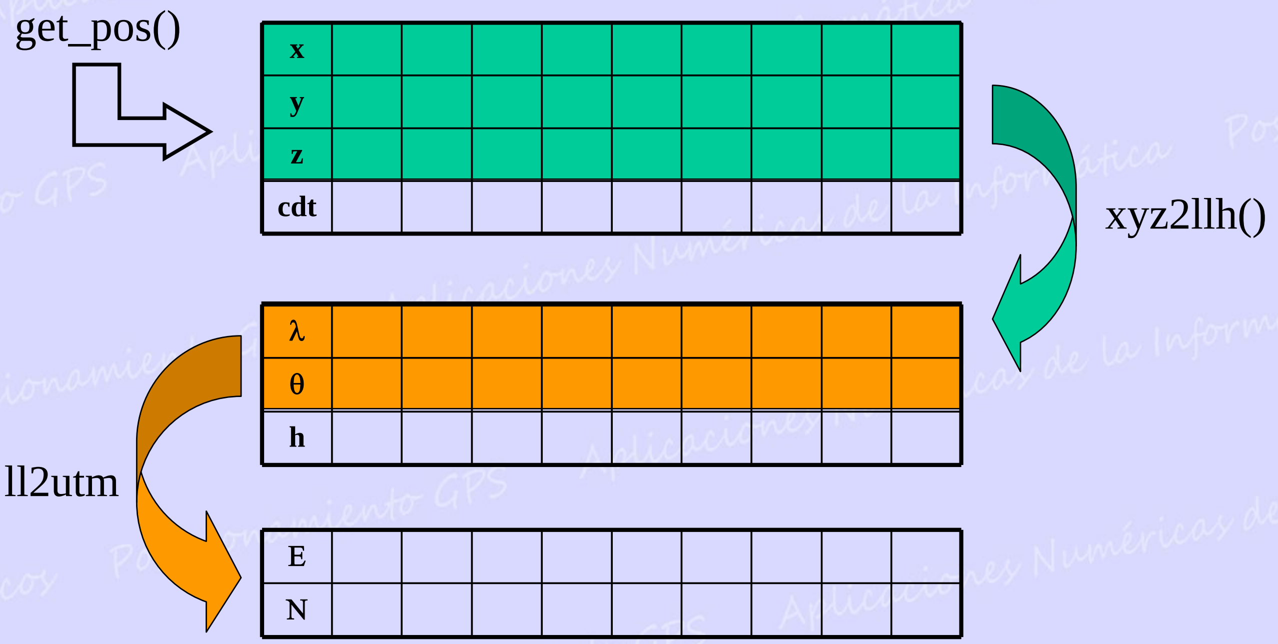
Proceso cambio de coordenadas







Proceso cambio de coordenadas



Dispersión de las medidas obtenidas.

En cualquiera de las coordenadas usadas estimaremos el error de las medidas obtenidas para ver si están en el rango de lo esperado (≅ 10's metros). Para ello se dispone de la posición exacta estacion.XYZ de la estación de referencia.

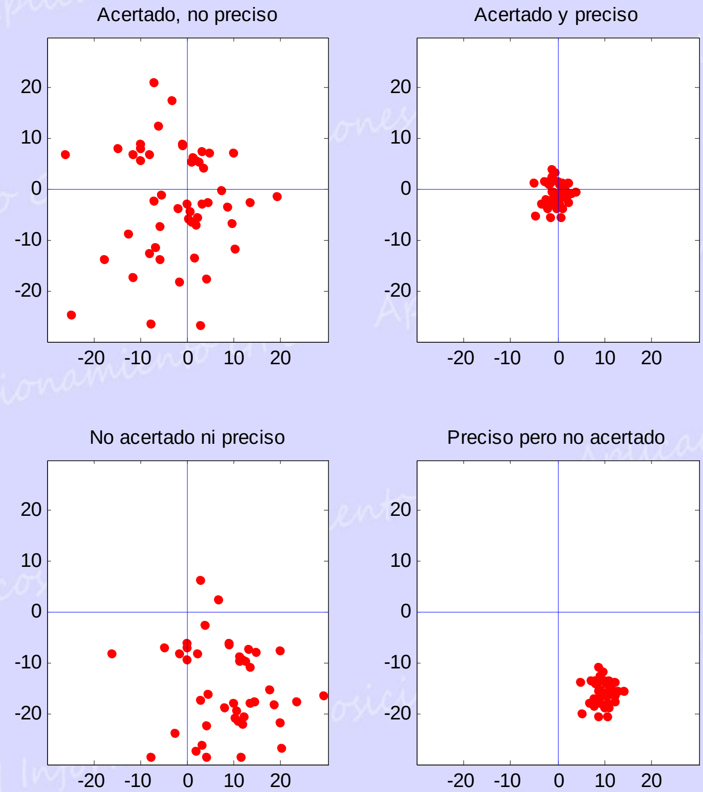
Haremos gráficas con las **diferencias** de nuestras medidas con la verdadera posición en las coordenadas (E,N) + H

- Aplicamos las funciones xyz2llh + ll2utm a la posición exacta (estacion.XYZ) podemos obtener la posición de la estación E0,N0,H0.
- Hallaremos la diferencia entre nuestras estimaciones (E,N,H) y las coordenadas exactas (E0,N0,H0) para obtener los errores dE,dN,dH.
- Mostraremos un gráfico 1D con los errores en altura (dH) y un gráfico 2D con las discrepancias en superficie (dE,dN)

Además de los gráficos estimaremos la dispersión de los errores calculando p.e. los errores máximos o la media de los errores en una u otra coordenada:

max(abs(dE)), mean(abs(dE))

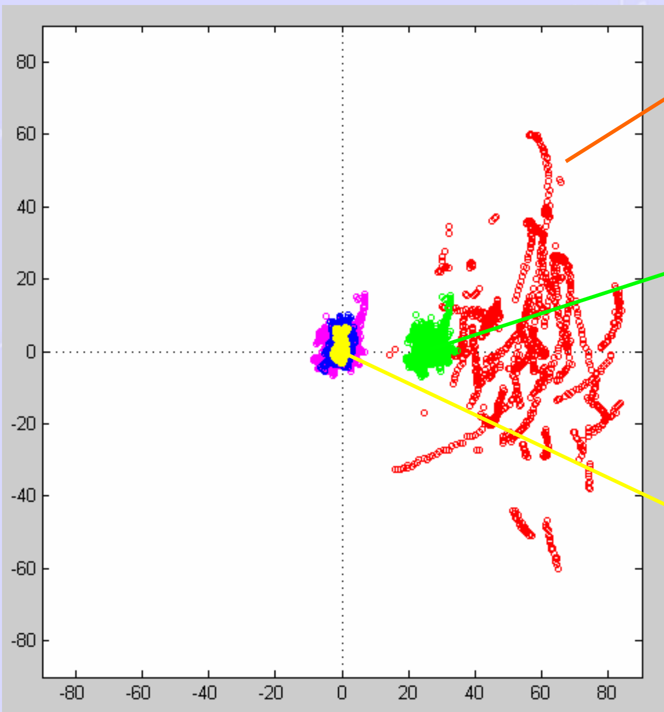
Acierto y Precisión



El **acierto** es una medida de la semejanza de una colección de medidas con la verdadera solución. Solo podemos calcularlo cuando se conoce la verdadera solución.

La **precisión** es el parecido de las medidas con su media, esto es, lo cerca que están unas de otras. Siempre es posible calcularla (desviación standard de los datos)

Errores aleatorios provocan una falta de precisión. Una falta de acierto suele reflejar un error o sesgo sistemático (tal vez un error en nuestro algoritmo o algún factor que no hemos considerado en nuestro modelo)



Poco acertado e impreciso

Más preciso pero todavía con poco acierto

Bastante acertado y preciso

Cuantificación del error

Es típico resumir los posibles errores en un solo número: p.e. 5 mt en horizontal y 10 en vertical. Pero, ¿es esa la media del error, el máximo error, ...?

Si no sabemos exactamente lo que significan estas cifras pueden ser engañosas.

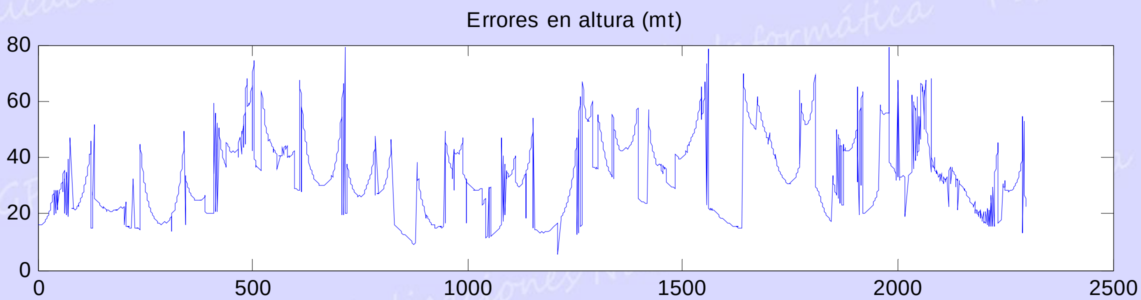
Normalmente dichos números reflejan lo que se denomina CEP50 (Circular Error Probability 50%). El CEP50 o simplemente CEP (Circle Equi Probable) es el radio de un circulo centrado en la verdadera posición (horizontal) dentro del cual caen el 50% de nuestras medidas. Otros valores más rigurosos usados son el CEP67 o CEP97 (círculos donde entran el 67% o 97% de la medidas).

Análogamente en 1D (al comparar alturas) se da un intervalo (+/-) alrededor de la altura correcta donde entra el % escogido de medidas (50%, 67%,...)

En 3D se usa el SEP (Spherical Error Probability): consiste en “inflar” una esfera centrada en la solución hasta que entren en ella el % adecuado de medidas.

Normalmente los fabricantes suelen citar los errores para una cifra del 50%, más optimistas que los del 67% o 97%, que son más usados en estudios más serios.

Resultados de un día (altura y superficie)

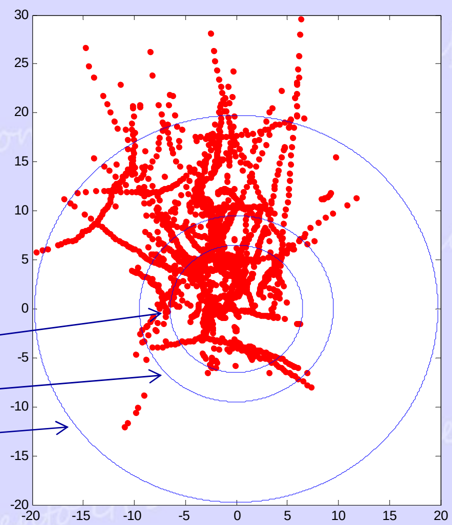


Error Altura. Media = 32.7 mt.
Desviación standard = 13.4 mt.

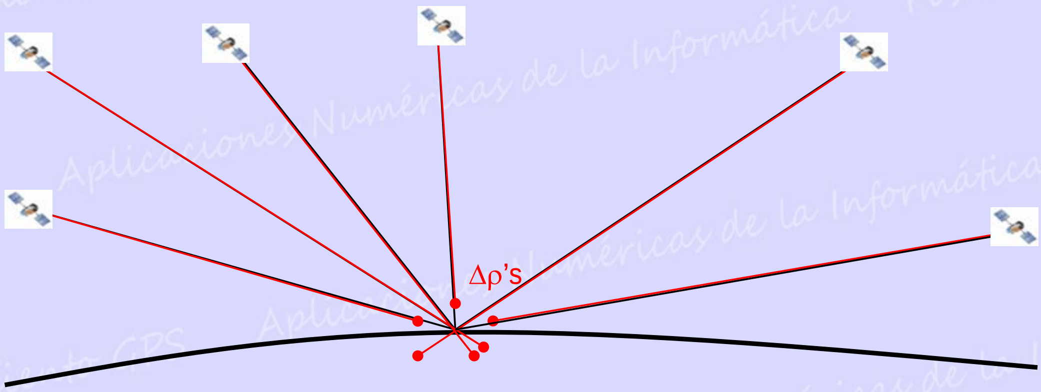
Error Este . Media = -2.3 mt.
Desviación standard = 4.1 mt.

Error Norte. Media = 5.9 mt.
Desviación standard = 6.3 mt.

CEP50 6.5m
CEP67 9.5m
CEP97 19.8m



Sistema sobredeterminado



Con más de 4 satélites obtenemos una solución matemática aproximada.

Al final de la iteración los residuos Δρ no se anulan por completo

Con nuestra solución el error se "reparte" por igual entre los residuos

Podemos indicar al algoritmo qué residuos queremos hacer más pequeños en función de lo fiables que sean las medidas correspondientes.

Mejoras: Uso de pesos

Con sólo 4 satélites no hay margen: debemos fiarnos al 100% de las 4 medidas, ya que las necesitamos todas.

Al observar más de 4 satélites lo que estamos haciendo es resolver un problema sobre-determinado, cuya solución trata de minimizar la expresión:

$$E = \sum_{k}^{SATS} \left(\rho_k^{obs} - \rho_k(\overline{x}) \right)^2 = \sum_{k}^{SATS} \Delta\rho_k^2$$

Al tener más satélites de los necesarios tenemos una cierta redundancia: podemos ser más selectivos y dar más importancia a unas observaciones que a otras.

Asignaremos una medida de la calidad Q>=0 a cada satélite, y la usaremos para ponderar sus residuos, haciendo un ajuste con pesos en el que minimizaremos:

$$E = \sum_{k}^{SATS} Q_k \cdot \left(\rho_k^{obs} - \rho_k(\overline{x}) \right)^2 = \sum_{k}^{SATS} Q_k \cdot \Delta\rho_k^2$$

Resolución sistema sobredeterminado con pesos

Matemáticamente es un problema de ajuste de datos con pesos, cuya solución es:

$$(H^T Q H)\cdot \Delta x = H^T Q \cdot (\Delta\rho) \Rightarrow \Delta x = \left(H^T Q H\right)^{-1} (H^T Q)(\Delta\rho)$$

donde Q es una matriz diagonal Nsat x Nsat cuya diagonal son los pesos {Q_k}.

Consideraciones:

- Los pesos no pueden ser negativos: unos errores no pueden cancelar a otros.
- Si todos los pesos valen lo mismo, la matriz Q es un múltiplo de la identidad, todos los satélites "valen" lo mismo y el resultado es equivalente a no usar pesos
- La solución hará más caso a satélites con Q alto, reduciendo más sus residuos (ya que contribuyen más al error global).
- Un valor Q $\cong$ 0 hace que la medida de ese satélite no contribuya a la solución.

Posibles criterios para Q: elevación sobre horizonte, calidad señal satélite, etc.

Función de calidad

Una típica elección es relacionar la calidad del satélite con su elevación φ sobre el horizonte: Q(φ) debe ser alto para satélites cerca del zenith y bajo para satélites cerca del horizonte (poco fiables, más atmósfera que atravesar, obstáculos, etc).

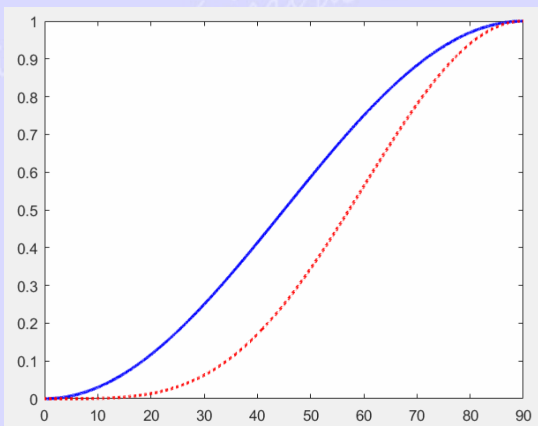
Una función sencilla que cumple esto puede ser: $Q(\varphi) = \dfrac{(1-\cos(2\varphi))}{2}$

La gráfica siguiente muestra la función Q frente a la elevación φ en color azul

Para elevaciones φ~0, cos(φ)~1 y Q~0. Por el contrario, si φ~90º, cos(180º)~-1 y Q~1

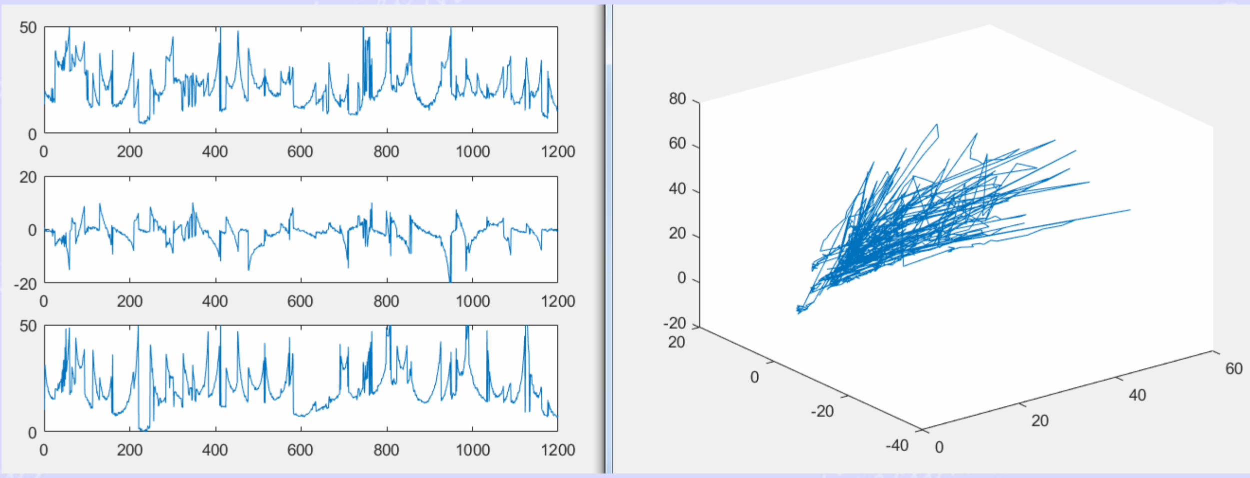
Para reducir más la influencia de los satélites más bajos se puede elevar Q al cuadrado (línea roja discontinua en la gráfica).

Si la elevación la tenemos en grados, recordad usar cosd() al calcular el coseno.

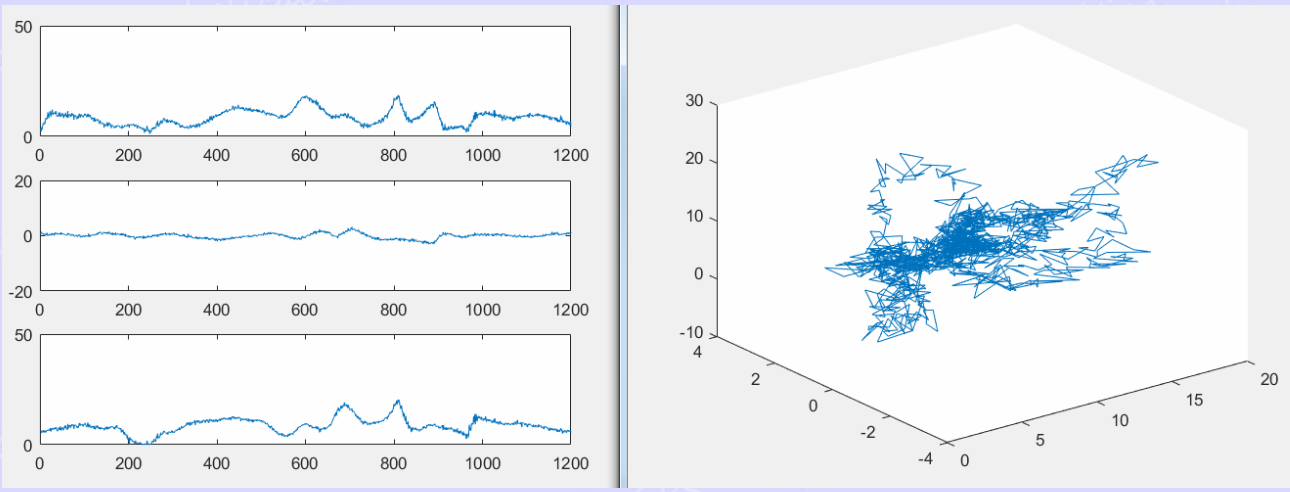


ANTONIO TABERNERO GALÁN, 2019 APLICACIONES NUMÉRICAS de la INFORMÁTICA, POSICIONAMIENTO GPS

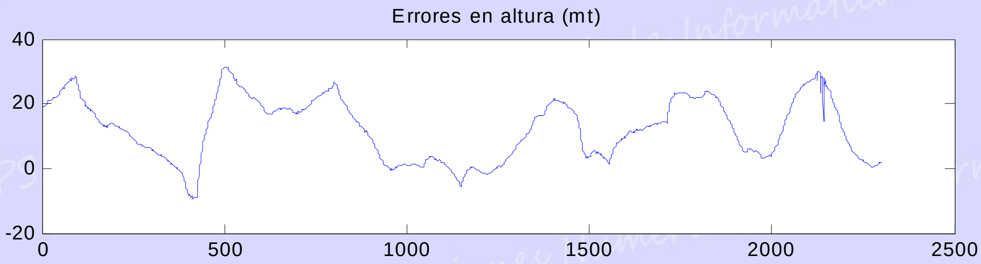
Resultados sin pesos (en XYZ)



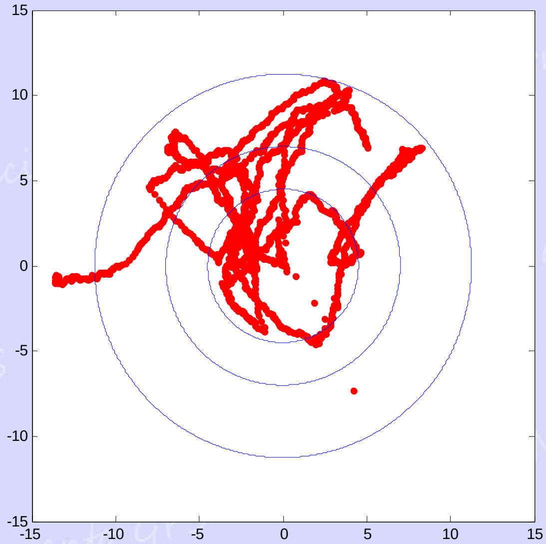
Resultados usando pesos (XYZ)



Resultados con pesos (altura y superficie)



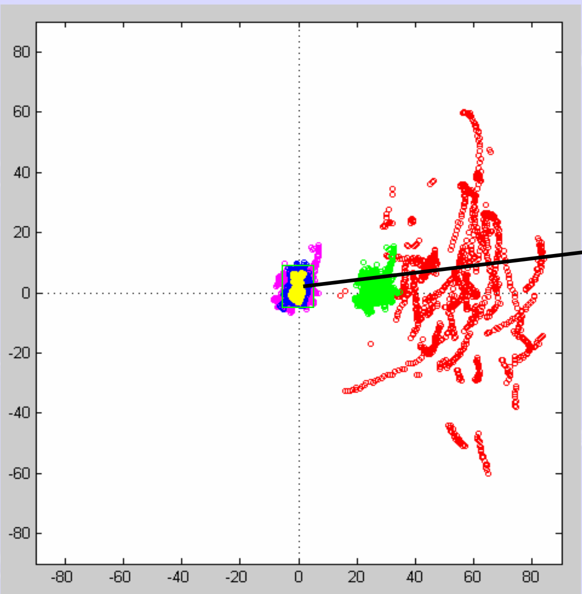

```
Error Altura. Media = 12.3 mt.
Desviación standard =  9.4 mt.


Error Este .  Media = -1.4 mt.
Desviación standard = 4.1 mt.

Error Norte.  Media = 3.1 mt.
Desviación standard = 3.7 mt.
```

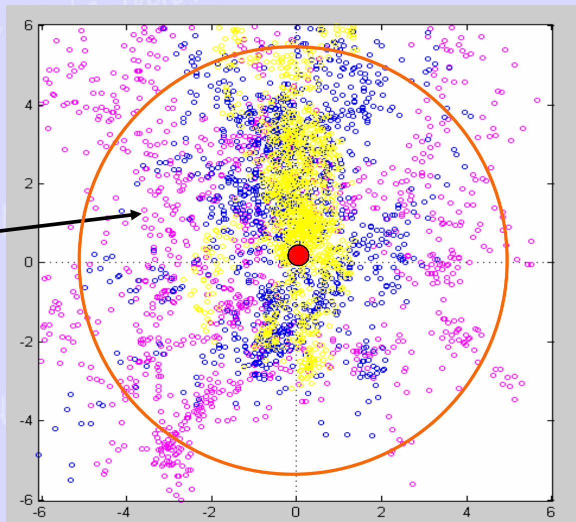



```
CEP50   4.5m
CEP67   7.0m
CEP97  11.3m
```

Posicionamiento GPS Diferencial

Posicionamiento Autónomo





¿ Podemos reducir aún más la dispersión observada (del orden de 4-5 m CEP95) ?

Precise Point Positioning (PPP)

Seguir con el enfoque explicado hasta ahora y mejorar la precisión a base de aumentar la complejidad del modelo

Dear GAMITeers,

svnav.dat and antmod.dat (igs08_1830_plus.atx) have been updated for the reassignment of PRN 26 to reactivated Block IIA SV 32 on 5 February.

Since the last use of PRN 26 (SV 26) was also a Block IIA SV, the radiation-pressure model will be the same.

However, the SV antenna offset is different by 7 cm.

With regards,

Dr. Robert W. King, Principal Research Scientist

Department of Earth, Atmospheric, and Planetary Sciences

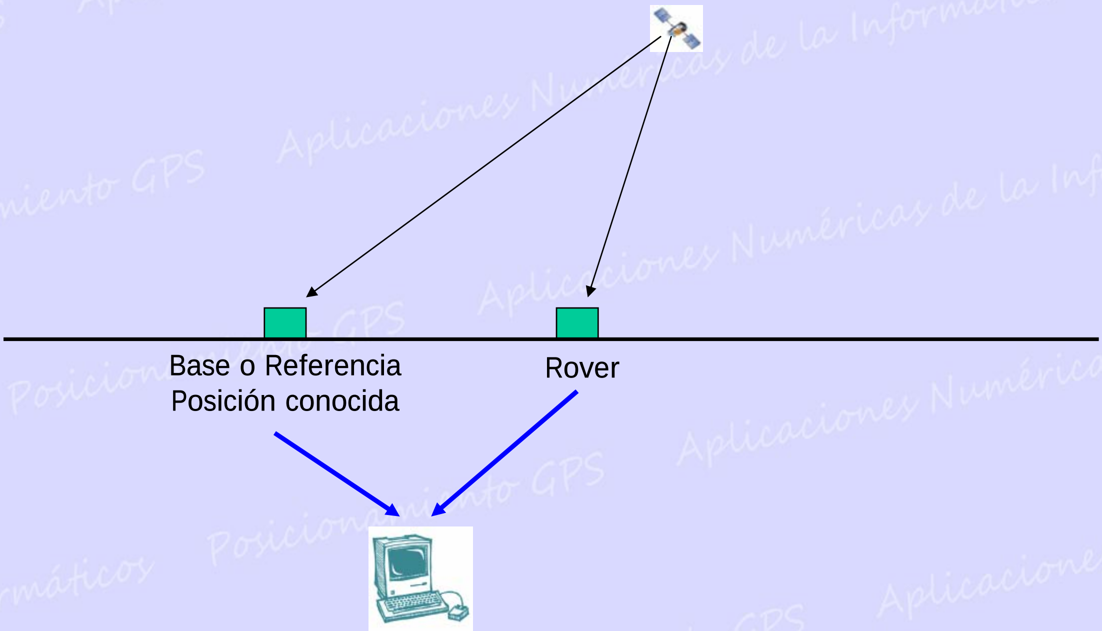
Massachusetts Institute of Technology

La otra opción: GPS Diferencial

- El error obtenido usando los algoritmos de posicionamiento autónomo (en buenas condiciones) es de unos 5-10 mt ($1\sigma-2\sigma$).
- Los distintos errores que contribuyen al error final se pueden clasificar dependiendo de dónde se originan:
  - **En los satélites: afectan a todo el sistema:**
    Errores en las efemérides, tanto en la posición/orientación del satélite como en los errores de los relojes.
  - **En la trayectoria entre satélite y receptor:**
    varían con la posición relativa receptor/satélite, tales como los retrasos ionosféricos y troposféricos.
  - **En el lugar de observación:**
    propios de cada receptor GPS (errores al medir tiempos de vuelo) y del lugar de observación (multipath en la zona de observación).
- Los errores con un origen común (tipo 1 y parte del 2) es posible reducirlos y/o cancelarlos comparando observaciones de dos posiciones relativamente cercanas: **GPS diferencial**

Diferencias Simples de Pseudo-rangos

Las DIFERENCIAS SIMPLES consisten en una resta de observables usando DOS ESTACIONES (referencia y rover) a UN MISMO satélite.



Modelo de diferencias simples de pseudorangos

Modelo de pseudo-rangos con observaciones simultáneas (mismo Tr):

rover: $\rho^K = R^K + dt(T_R) - dT^K(T_R - \tau^K) + I^K + T^K$

\- ref: $\rho^K_{ref} = R^K_{ref} + dt_{ref}(T_R) - dT^K(T_R - \tau_{ref}{}^K) + I_{ref}{}^K + T_{ref}{}^K$

$$(\rho^K - \rho^K_{ref}) = (R^K - R^K_{ref}) + (dt(T_R) - dt_{ref}(T_R))$$

- Para posiciones **suficientemente cercanas** los errores del satélite (pos, dT), y del viaje (ionosfera / troposfera) serán muy parecidos y se cancelaran al restar ambas ecuaciones (diferencias simples):
- Los errores de reloj de los receptores no son iguales y no se cancelan, pero se agrupan en una sola incógnita Δt: $(\rho^K - \rho^K_{ref}) = (R^K - R^K_{ref}) + \Delta t$
- Cuatro incógnitas (x,y,z y Δt) → necesitamos un mínimo de 4 sats comunes para poder calcular nuestra posición usando GPS diferencial.

Resolución del modelo de GPS diferencial

- Similar al caso autónomo (más sencillo al ignorar ciertos errores)
- La matriz H (derivadas de las diferencias de pseudorrangos respecto a las incógnitas) es la **misma matriz H** del caso autónomo:

$$H = \begin{pmatrix} -u_X^1 & -u_Y^1 & -u_Z^1 & 1 \\ -u_X^2 & -u_Y^2 & -u_Z^2 & 1 \\ -u_X^3 & -u_Y^3 & -u_Z^3 & 1 \\ \cdots & \cdots & \cdots & \cdots \\ -u_X^m & -u_Y^m & -u_Z^m & 1 \end{pmatrix}$$

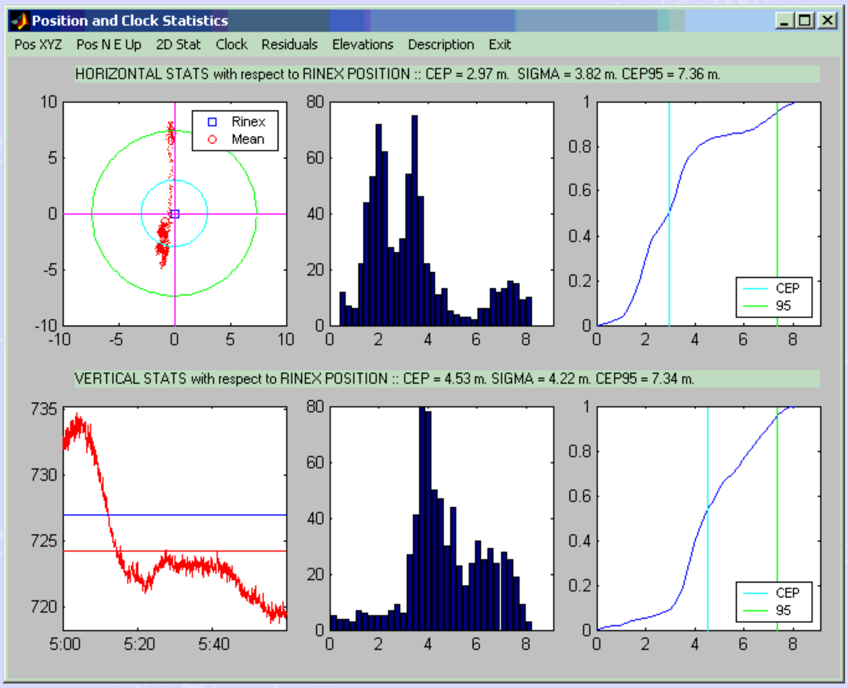
- Misma ecuación a resolver, pero ahora trabajamos con diferencias de pseudo-rangos en vez de pseudo-rangos como en el caso anterior:

$$\underbrace{(\overline{\rho}_{rov} - \overline{\rho}_{ref})_{obs}}_{\Delta\text{Medidas}} - \underbrace{(\overline{\rho}_{rov} - \overline{\rho}_{ref})_{esp}}_{\Delta\text{Predicciones}} = H \cdot \overline{\delta x} \qquad \overline{x} = \overline{x} + \cdot \underbrace{\overline{\delta x}}_{\text{Corrección}}$$

Sincronización de observaciones

- En el modelo se asume que el tiempo de recepción Tr es común para ambos receptores: ¿cómo podemos sincronizar dos GPS?
- Sabemos que GPS sirve también como un sistema global de tiempos: una vez “enganchados” cualquier receptor en cualquier lugar está sincronizado con el tiempo GPS (con una precisión de fracciones de μsec).
- La norma es que todo receptor GPS que se precie tome sus medidas cerca de un segundo GPS exacto, para eventualmente poder comparar sus observaciones con las de otro GPS (post-procesado).
- ¿Cómo de crítica es la sincronización necesaria? Según los observables.
  - Pseudorangos: podemos permitirnos un desfase del orden de un milisegundo antes de que el error introducido por este desfase supere el error debido a otras fuentes.
- Si el desfase es mayor de 1 msec (o si trabajamos con observables más críticos) deberíamos “llevar” las observaciones a un instante de tiempo común antes de restarlas. Esto puede hacerse a través de interpolación.

Resultados DGPS pseudorangos (MER1 / MER2)



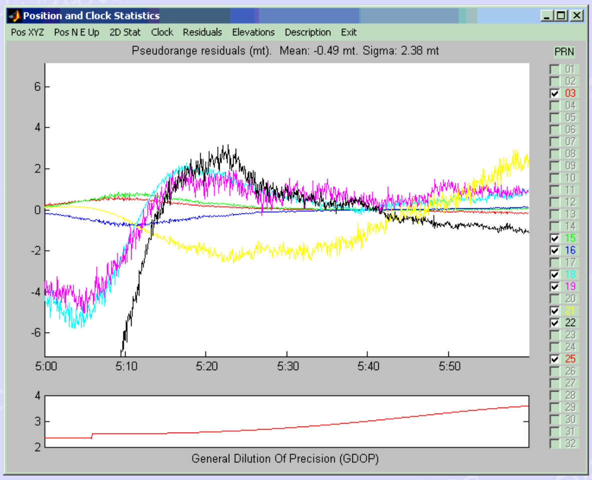
<u>Caso Ideal</u>

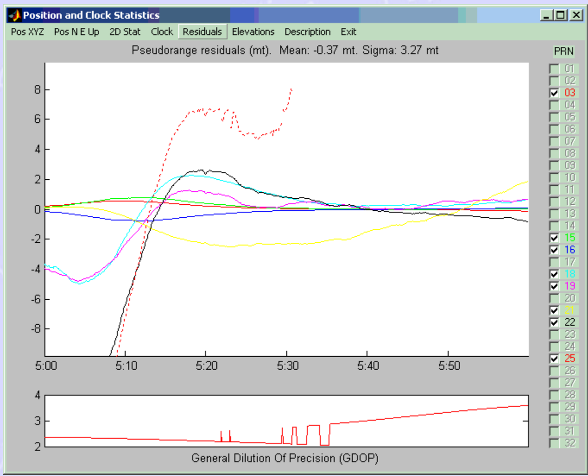
- Dos buenos GPSs (Trimble y Leica) situados a unos 10 metros entre ellos.
- Resultados para estación MER1 procesada de <u>forma autónoma</u>.
- Errores máximos de unos 8-10 m.

CEP50 = 2.97 m

CEP95 = 7.35 m

Residuos del procesado autónomo de MER1 y MER2)

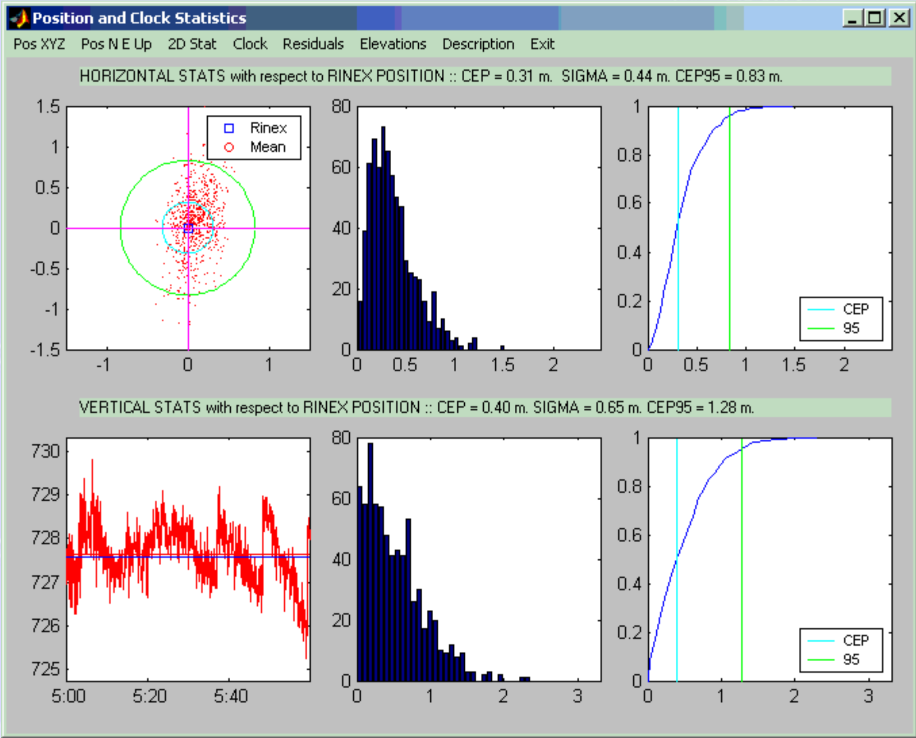




Izquierda: residuos entre pseudorangos observados/esperados en MER1
Derecha: mismos resultados para la estación MER2.

- Su similitud indica que se deben a factores externos.
- Esperamos que se cancelen al restar las observaciones → buena solución en modo DGPS.

Procesado DGPS pseudorangos (MER1 / MER2)



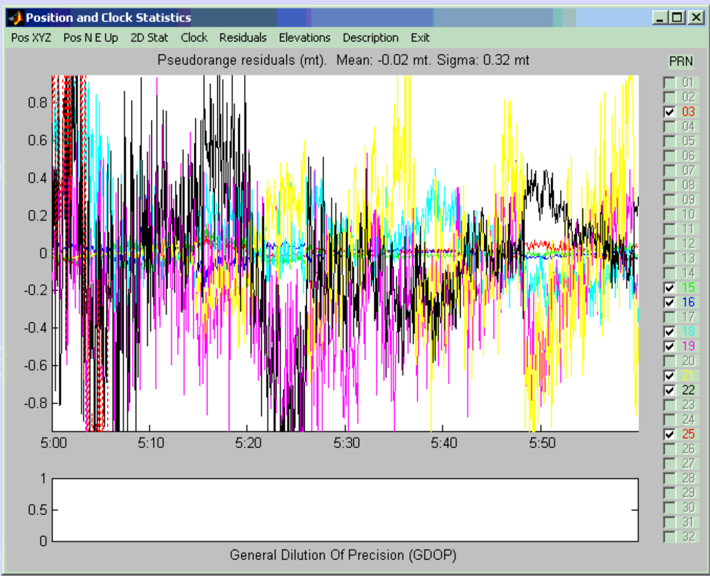
- Resultados del procesado DGPS de pseudorrangos entre las estaciones de MER1 y MER2. (superficie y altura).
- Reducción de la dispersión: errores máximos de menos de 1.5m. (horizontal) y unos 2m. (altura). Varianza por debajo del metro en ambos casos.

En superficie:

CEP50 30 cm

CEP95 85 cm

Residuos del procesado DGPS



- Pasamos de una σ en los residuos de unos 2/3 mt. a sólo 0.3 metros.
- Errores máximos del orden de +/- 1metro.
- La varianza residual refleja los ruidos de medida de ambos aparatos.