Bloque I: Posicionamiento GPS

OBJETIVOS:

1. **Comprender el funcionamiento básico del sistema GPS, presentando un esquema simple del modelo de posicionamiento con algunas (no todas) de sus complicaciones.**

2. **Implementar las técnicas numéricas para solucionar el problema:**
   - **Interpolación: nodos equi-espaciados / diferencias finitas.**
   - **Resolución de sistemas de ecuaciones no lineales: método de Newton en varias dimensiones.**
   - **Ajuste de datos (sistemas lineales sobredeterminados).**

OBJETIVO FINAL: reproducir el software que usa un GPS para darnos nuestra posición con una precisión de unos 5-10 m.

REFERENCIAS

Las charlas serán "autocontenidas", pero si os interesa profundizar:

- **"GPS, Theory and Practice"**

  Hofmann-Wellenhof et al. , 4th Edition, 1997 Springer, New York.
- **Especificación oficial de la señal GPS:** (google gpssps1.pdf)

- **"Global Positioning System: Theory and Applications"**

  E.W. Parkinson, J.J. Spilker, Editors, Vol. I and II. Progress in Astronautics and Aeronautics, vol 163-164. American Institute of Aeronautics and Astronautics, Inc.
- **"GPS Satellite Surveying",**

  Alfred Leick, 3th Edition, 2004, John Wiley & Sons, New Jersey
- **"A Software-Defined GPS and GALILEO receiver"**

  Borre et al., Birkhäuser, Boston, 2007

Descripción del sistema GPS (GNSS)

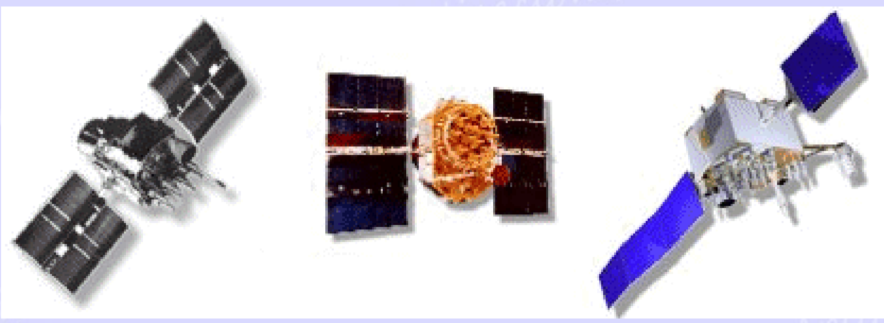
- GPS (Global Positioning System) es un sistema de radio-navegación con cobertura global, 24 horas al día, en toda situación meteorológica.
- Depende del Departamento de Defensa (DoD) de los EEUU.
- Oficialmente NAVSTAR (NAVigation Satellite Timing and Ranging).
- A veces se usa el término GNSS (Global Navigation Satellite System) que engloba a otros sistemas similares actuales (GLONNASS ruso) o futuros (GALILEO europeo).

- El sistema GPS se divide en tres partes:

  1. Segmento espacial: los satélites.
  2. Segmento de control: centros de control y mantenimiento.
  3. Segmento de usuario: nuestros receptores GPS.

Segmento espacial

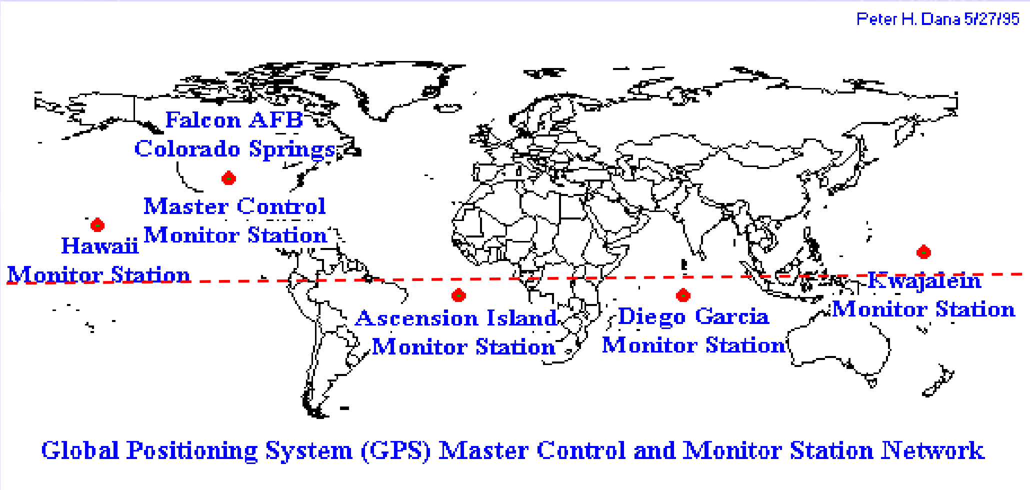
- Compuesto por los satélites GPS y las cohetes lanzaderas (Delta) que los ponen en órbita (a unos 20.000 km de altura, la mitad de una órbita geo-síncrona).
- Hay unos 28-32 satélites operativos, garantizándose que al menos 6 serán visibles simultáneamente desde cualquier lugar.
- Hay varias generaciones de satélites en el "aire" (bloque I, II, IIR), cada uno con diferentes funcionalidades. En la imagen se muestran ejemplos del bloque I (izquierda), II (centro) y IIR (derecha).



Segmento de control

• Estación principal: una base aérea en Colorado (USA).

• Varias estaciones de monitorización en diversas islas.

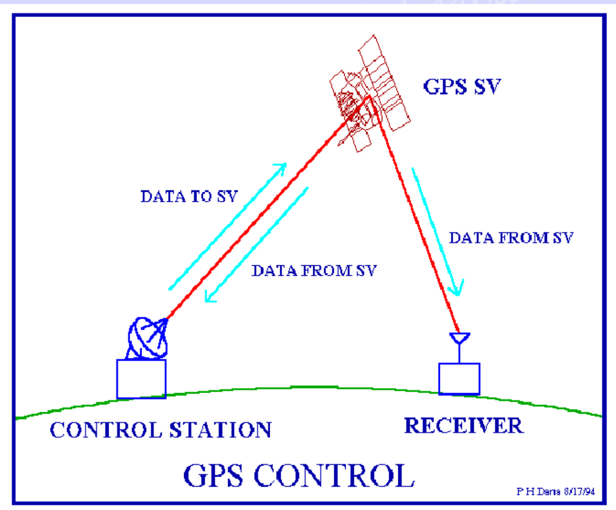


• **Salvo la estación principal, las demás están próximas al Ecuador.**

Segmento de control

• Las estaciones de seguimiento monitorizan los satélites GPS y mandan los datos recogidos a la estación central en Colorado, donde se estiman sus posiciones en las siguientes horas en forma de parámetros orbitales.

• Dicha información se envía a los satélites para que a su vez éstos se la reenvíen a los usuarios GPS en el llamado **mensaje de navegación**.

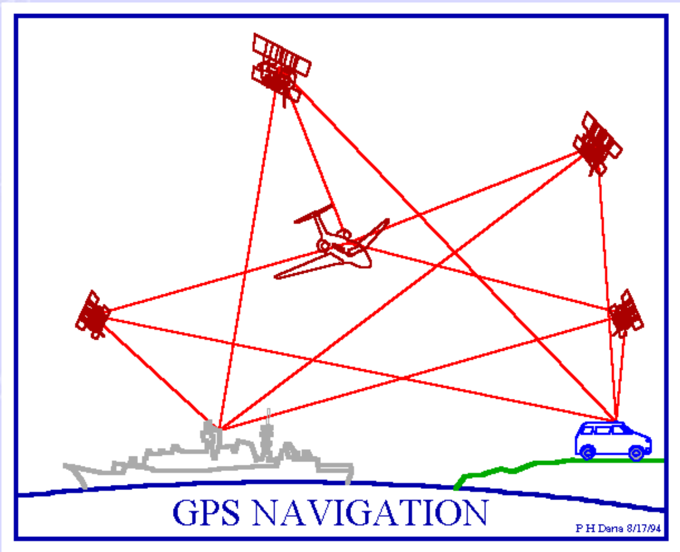


• El segmento de control es el único autorizado a "hablar" con los satélites, los demás usuarios sólo "escuchamos".

• GPS es un sistema PASIVO de localización, lo que permite un número ilimitado de usuarios.

Segmento de usuario

- Los receptores GPS captan las señales enviadas por los satélites y decodificar su contenido, encontrando su posición a través de un proceso que estudiaremos en detalle.

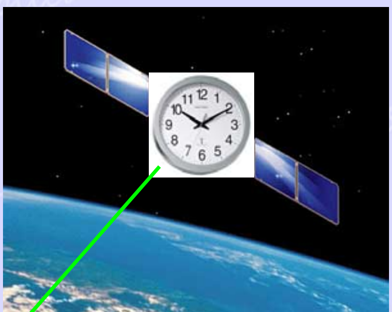


- Se trata de medir la distancia a varios satélites y plantear un modelo que relaciona nuestra posición con dichas distancias.
- Solucionando las ecuaciones resultantes hallamos nuestra posición.
- El usuario no necesita hablar con los satélites, solo escuchar

Fundamentos del posicionamiento GPS

Basado en la medida del tiempo de vuelo de la señal entre satélite-receptor.

Imaginad un reloj bien visible en cada satélite GPS.



Tsat = 10:10:37.060

Trec = 10:10:37.127

τ= 0.067 segs

c = 299792458 m/seg

c·τ = 20086.095 Km

Simultáneamente el observador mira al reloj del satélite y al suyo.

Los relojes no marcan lo mismo porque el usuario ve la hora del reloj cuando la luz salió del satélite (tiempo de transmisión Tx).

En el reloj local vemos la hora de llegada (tiempo recepción Tr).

La diferencia observada τ=(Tr -Tx) sería el tiempo de vuelo.

Multiplicando por la velocidad de la luz c determinamos la distancia.

¿Cómo se hace esto realmente?

En la señal GPS se inserta un código pseudo-aleatorio (PRN), con 1023 “chips” (0’s y 1’s) que se repite a intervalos regulares (cada milisegundo).

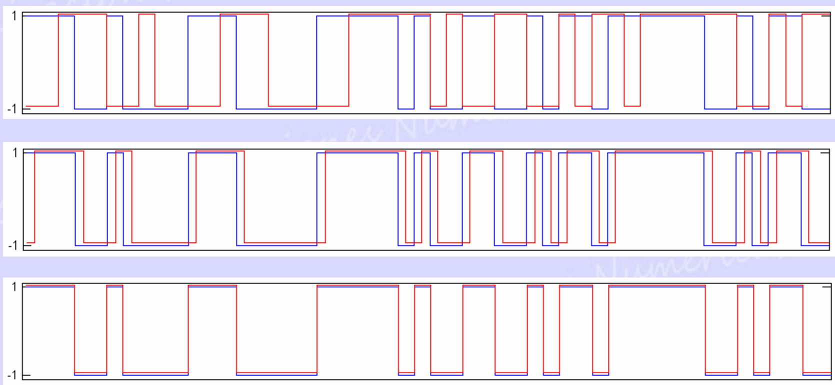
Cada satélite se identifica con un PRN propio. Por ejemplo el PRN1 es:

1100100000111001010010011110010100010011111010101101000100010101010110010001111010011111101101110011011111
0010101010000100000000111010100100010011011110000011110101110011001111011000000010111100111110101001100010
1101110001101111010100010101100000100000000100000011000111011000000111000110111111111010011101001011011000
0101010110001001110010110111011000111011101111000011011000011001001001000001101101001011011110001011100000
0101001001111110000010101011100111110101111100110011000111000110110101010110110001101110111000000000001011
0011011001110110100000101010111010111010010100011100111000100101000101001011010000101011011010110110001110
0111101100100001111110010110100010000111110101011100110010010010010111111111110000111110111100011011100101 1
0000111001010100001010010101111110001111011010011101100111111011111010001100011111000000001001010001011010 0
0100010011011000000111011010001101000100100011100010110011001001111001101111111001100101001101001101011110 0
110110101001110111100011010100010000100010010011100000111001010001 0000

El receptor reproduce estos PRNs (cuya especificación es pública) e intenta alinearlos con los que nos llegan en la señal del satélite.

¿Qué precisión podemos esperar?

El retraso que el receptor debe introducir en el PRN generado (rojo) para estar alineado con el que nos llega del satélite (azul) nos permite estimar el tiempo de vuelo y la distancia al satélite.



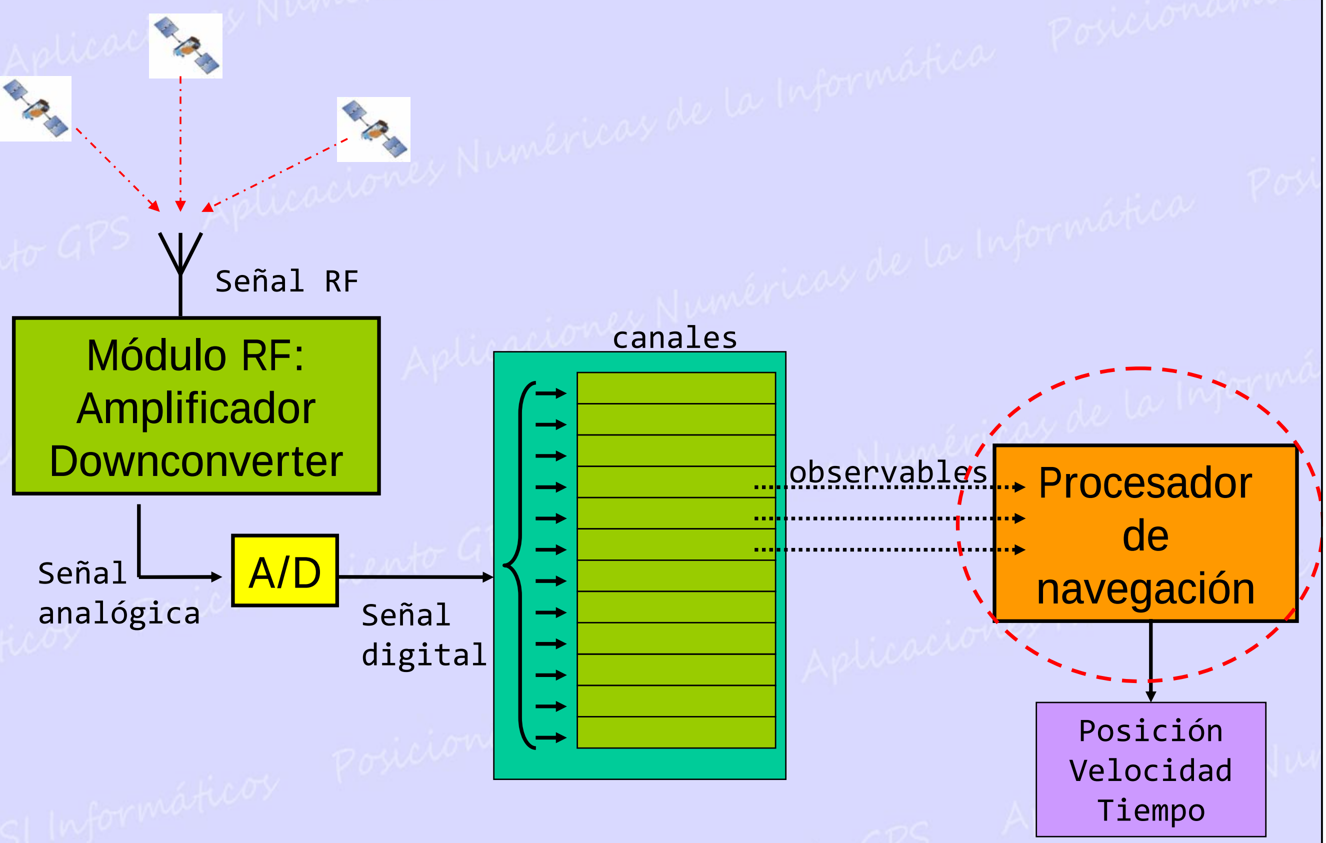
Los PRNs están diseñados para que su alineación sea muy crítica.

Es posible alinearlos con una precisión del orden de un 1% de un chip

Un chip dura aproximadamente 1 μs = 300 m. 1% de eso son unos 3 m.

De ese orden es la precisión de las medidas de distancias a los satélites.

Subsistemas en un receptor GPS



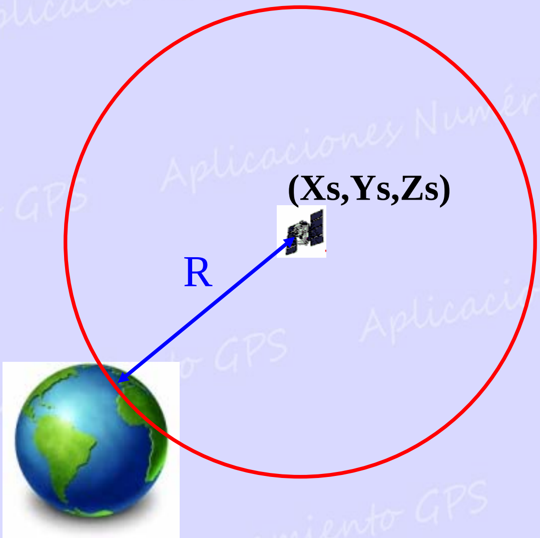
Subsistemas en un receptor GPS

Un receptor GPS está dividido en varios subsistemas funcionales:

1. **Módulo de RadioFrecuencia (RF) que comprende la antena, un amplificador y un demodulador para bajar la frecuencia de la señal GPS (1475 Mhz) a algo más manejable, seguido de un conversor analógico digital (A/D) que muestrea la señal resultante a un ritmo de unos pocos MHz. La salida es un *stream* de varios millones de muestras por segundo.**
2. **Un procesador especializado que con técnicas de Procesado Digital de Señal es capaz de obtener información sobre tiempos de vuelo de la señal, extraer los bits del mensaje de navegación, etc.**
3. **Finalmente un procesador de carácter general recibe dicha información (distancias a satélites + mensaje navegación) y a partir de ella determina la posición del GPS.**

Es este último subsistema el que vamos a reproducir en este curso.

Posicionamiento GPS (simplificado)



- El satélite emite un PRN en tiempo Tx
- El receptor recibe el PRN en tiempo Tr
- El tiempo de vuelo es τ = (Tr-Tx)
- Multiplicando τ por la velocidad de luz c tendríamos R=distancia satélite-receptor.

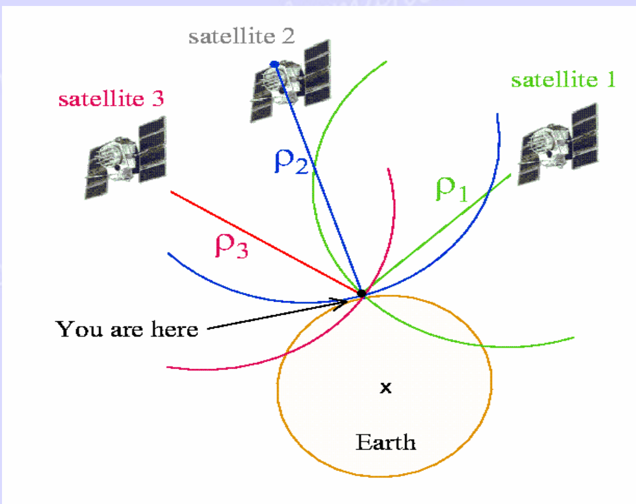
- **<u>Si conocemos</u>** la posición del satélite (Xs,Ys,Zs) podemos plantear la ecuación:

$$R = c \cdot \tau = \sqrt{(X_S - x)^2 + (Y_S - y)^2 + (Z_S - z)^2}$$

La posición del receptor (x,y,z) se encuentra en algún punto de la esfera (de radio R) centrada en la posición (conocida) del satélite (Xs,Ys,Zs).

Número ecuaciones = Número incógnitas

En el espacio 3D la intersección de tres esferas nos permite fijar nuestra posición (xyz)



- En general, para poder resolver el problema se precisa que el número de ecuaciones (observaciones) del modelo >= número de incógnitas.
- Con 3 incógnitas (x,y,z) precisaremos 3 ecuaciones (3 satélites).

Esquema general

- La realidad es más complicada, pero el enfoque general es válido:

1) Un receptor GPS toma una serie de medidas (observables, típicamente distancias a los satélites).

2) Establecemos un modelo entre los observables y las incógnitas (típicamente nuestra posición, aunque pueden ser otras)

3) Consideramos una hipótesis inicial para las incógnitas y aplicamos el modelo, calculando los observables esperados

4) Comparamos los observables predichos con los medidos.

5) Usamos la discrepancia entre lo Observado (medidas) y lo Esperado (modelo) para corregir la hipótesis inicial.

6) Repetimos pasos 3)-5) hasta la convergencia: el resultado del modelo coincide (más o menos) con nuestras observaciones.

En el caso descrito los observables serían las distancias R a los satélites, las incógnitas nuestra posición (xyz) y el modelo la ecuación de la esfera.

Revolución GPS (Posicionamiento)

- Aplicaciones de localización: Control de flotas, cartografía digital, navegación, ocio, excursionismo, navegación. (precisión ~m)
- Topografía, obra civil: monitorización de estructuras, trazado carreteras, catastro, etc. (precisión ~cm)
- Geodesia: monitorización de movimiento de placas, sismos (~mm)

Revolución GPS (Tiempos)

- GPS se ha convertido en un standard global de difusión de tiempo.
- El sistema GPS está basado en la medida de tiempos, no distancias.
- Un GPS puede conocer su posición con una precisión de unos 5/10 m. Para ello es preciso conocer el tiempo con una precisión comparable:

$\Delta x = c \cdot \Delta t$ con c = 299792458 metros por segundo

$\Delta x = 6$ metros en distancia equivalen a $\Delta t = \Delta x / c = 20$ nanosegundos

Sistemas de Referencia Espacial

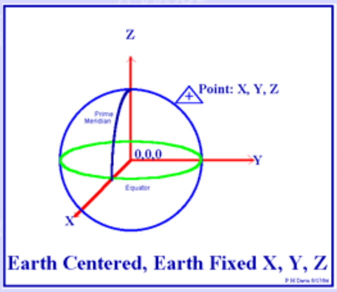
- Hemos hablado muy alegremente de tiempos Tx,Tr y posiciones X,Y,Z
- Si queremos llegar a la precisión de la que estamos hablando hay que especificar MUY BIEN el sistema de referencia usado.

GPS usa un sistema de referencia XYZ con los ejes centrados y solidarios con la Tierra (ECEF, Earth Centered Earth Fixed)

Z -> Sale por el Polo Norte

X -> Sale por el Ecuador, por el meridiano de Greenwich

Y -> También por el Ecuador, formando sistema dextrógiro



Earth Centered, Earth Fixed X, Y, Z

En GPS usamos sistema de referencia WGS84 (World Geodetic Survey)

Sistema referencia temporal

Los tiempos se expresan en el llamado tiempo GPS, que está determinado por varios relojes atómicos en el centro de control.

El origen del tiempo GPS es la medianoche entre sábado 5 y el domingo 6 de enero de 1980 (6/01/1980 a las 00:00h).

El tiempo GPS se expresa en semanas (week) desde dicho arranque y en los segundos dentro de la semana (tow, time of week)

El origen de la semana es la medianoche entre sabado y domingo, luego el primer día de una semana GPS es el domingo.

El tow va desde 0 a 604799 (número de segundos en una semana):

01:00:00, Enero 6, 1980	->	Semana: 0	TOW: 003600
23:59:59, Enero 12, 1980	->	Semana: 0	TOW: 604799
01:00:00, Enero 13, 1980	->	Semana: 1	TOW: 003600
10:00:00, Sept 7, 2018	->	Semana: 2017	TOW: 468000

Posiciones de los Satélites GPS

- **Saber nuestra posición (x,y,z) exige saber la de los satélites (Xs,Ys,Zs).**
- **Nuestra primera tarea es saber obtener la posición de un satélite dado en cualquier instante de tiempo.**
- **Datos de partida: ficheros SP3 publicados por organismo internacional, con las posiciones XYZ (referidas al centro de la Tierra) de todos los satélites operativos a intervalos de 15 minutos (900 segundos).**

- **<u>NO ES ASÍ</u> como un receptor GPS autónomo consigue datos de los sats.**
- **Los datos de los satélites que llegan al GPS son emitidos por los propios satélites y vienen en forma de parámetros orbitales (semiejes de órbitas, excentricidad, inclinación, precesión, etc.)**
- **Aunque menos precisos, estos datos orbitales tienen la ventaja de que son válidos durante más tiempo (6-8 horas), disminuyendo la frecuencia con la que se deben "renovar" los datos.**

Estructura fichero SP3

Comienza con una cabecera con algunos datos generales y luego va dando las posiciones de los satélites a lo largo de todo un día.

En una línea (*) nos da la hora y en las siguientes (P) las posiciones XYZ (en km) y error de reloj de todos los satélites operativos (μsec)

```
*  2001 12 30  0  0  0.00000000
P  1  25566.266161    767.616279   7401.913077    209.426857
P  2  16054.036240 -10713.672719 -17368.462332   -128.710799
P  3   7320.779633  14311.262085 -21203.954293     96.437901
…
*  2001 12 30  0 15  0.00000000
P  1  24654.091403   1170.941513  10031.456875    209.429657
P  2  15774.839667  -8503.909842 -18777.770375   -128.715784
P  3   4981.379816  14990.508926 -21401.928030     96.439610
…
```

Tiempos (cada 15 min)

ID sat

Posición X,Y,Z del sat (en km) respecto al centro de la Tierra.

Error reloj (en μsec)

El nombre de un fichero SP3 indica la semana y día de la semana GPS:

`IGS12651.SP3  -> semana GPS 1265, 2º día de la semana (lunes)`

Función read_sp3

Os doy una función para leer estos ficheros: `sp=read_sp3('fich.SP3');`
que lee datos de 'fich.sp3' y devuelve una estructura sp, con campos:

```
sp.week:  semana GPS
sp.delta: salto de tiempo entre datos (normalmente 900 sec)
sp.nsat:  número de satélites para los que hay información.

sp.prn : tabla de 1 x nsat con los ID de los satélites.
sp.tow : tabla de 1 x 96 con los tiempos de los datos. Los tiempos se
         dan en segundos dentro de la semana (tow = time of week).
         Como un fichero cubre un día tenemos 24*60/15 = 96 tiempos.

sp.XYZ: posiciones sats: tabla de 3(XYZ) x 96 (tiempos) x nsat (#sats)
sp.cdT: errores de reloj de sats: tabla de nsat (#sats) x 96 (tiempos)
```

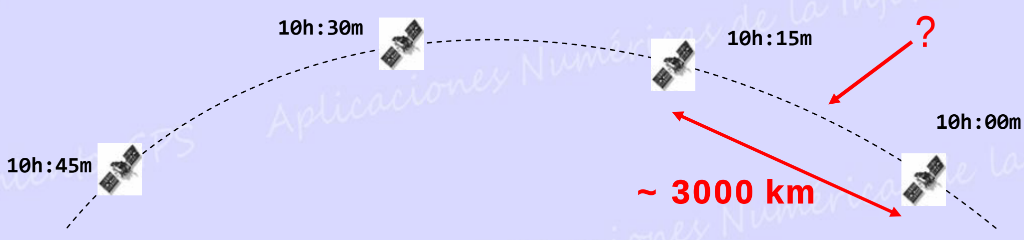
Por ejemplo, sp.XYZ(3,2,7) es la 3ª coordenada (Z) en el instante de tiempo tow=sp.tow(2) para el satélite con PRN = sp.prn(7)

sp.cdT(7,:) son los errores de reloj del satélite con PRN=sp.prn(7) a lo largo de todo el día.

En la estructura sp tanto las posiciones XYZ como el error de reloj (cdT) se dan CONVERTIDOS a metros (en el caso del error de reloj se usa la velocidad de la luz c = 299792458 m/s para pasar de μsec a metros).

Interpolación de la posición de un satélite.

Objetivo: determinar la posición de un satélite en un tiempo t dado, a partir del fichero SP3 con los datos del día correspondiente.

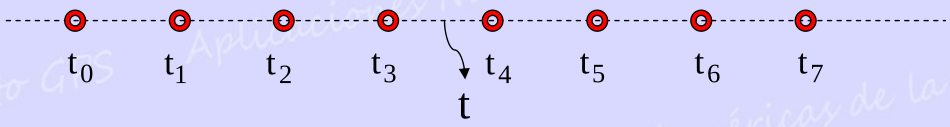


Pega: sólo tenemos datos cada 15 minutos y obviamente queremos poder resolver nuestra posición en cualquier momento.

Solución: usar los datos conocidos (fichero SP3) como una tabla de interpolación e interpolar las posiciones intermedias para el tiempo t.

¿Cuántos datos usar en la interpolación?

Lo usual es decidirse por un número par de nodos N, de forma que la mitad (N/2) caigan delante (o encima) del tiempo t y la otra mitad (N/2) detrás. Un valor típico para el número de nodos a usar es N=8:



En nuestros programas N será una variable para que podamos modificarla y observar los cambios en los resultados.

Dado un tiempo t, lo primero es determinar los N/2 nodos anteriores y los N/2 posteriores a dicho tiempo del fichero SP3 y construir con ellos una tabla de N datos, que será el punto de inicio del problema de interpolación.

Hallar los nodos a usar

PROBLEMA: dado un tiempo t y un fichero SP3 que lo contiene, determinar los N/2 nodos anteriores y posteriores al tiempo t.

El primer instante de tiempo del fichero es sp.tow(1), por lo que para encontrar el nodo inmediatamente anterior (prev) al tiempo t haremos:

```
x=(t-sp.tow(1))/sp.delta; prev=floor(x)+1;
```

Si t cae entre los dos primeros nodos, x valdrá 0.?? y prev valdrá 1, indicando que t está entre el 1er y el 2º nodo (los índices de MATLAB empiezan en 1).

Los índices de los N nodos (N/2 delante y N/2 detrás de t) a usar serán:

```
rg = (prev-N/2+1:prev+N/2);
```

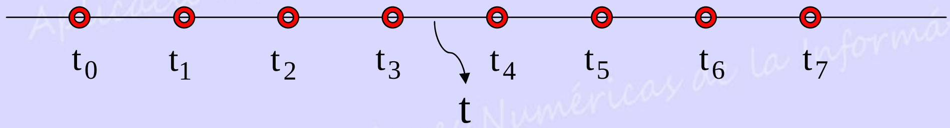
A partir de estos índices extraemos los valores para la tabla de interpolación:

`F = sp.XYZ(1,rg,2)` → coords X (1ª) del 2º sat en los nodos (rg) dados.

`F = sp.cdT(2,rg)` → error de reloj del 2º sat en los nodos (rg) dados.

Resumen

A) A partir de t hallar los N nodos (N/2 anteriores y N/2 posteriores) que enmarcan el valor de t. Por ejemplo para el caso de N = 8 nodos:



B) Extraemos la correspondiente información de los nodos escogidos y la usamos para construir una tabla de interpolación.

Para interpolar la posición X del 3er satélite de la lista extraemos las 8 X(k) correspondientes a los nodos t(k) dados obteniendo la tabla:

```
tk = sp.tow(rg)
xk = sp.XYZdT(1,rg,3)
```

→ tiempos:
→ coords X

t_0	t_1	t_2	...	t_7
X_0	X_1	X_2	...	X_7

C) Usar la tabla anterior para interpolar el valor de X para el tiempo t.

D) Repetiremos (mismos tiempos) para los valores de Y, Z y cdT

Fórmula de Newton y diferencias divididas

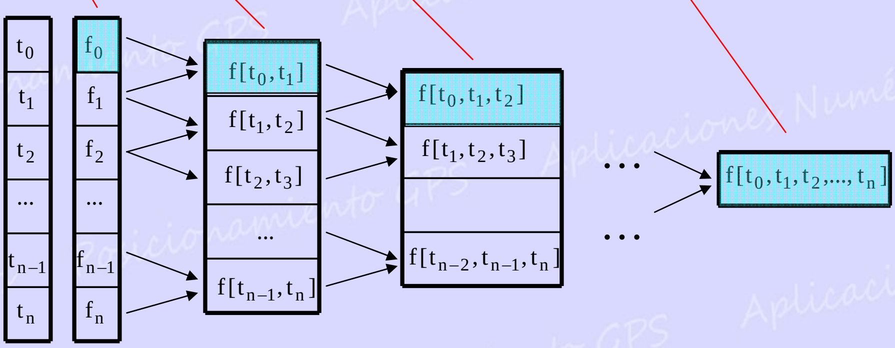
Tabla de datos a interpolar:

t_0	t_1	t_2	...	t_{N-1}
f_0	f_1	f_2	...	f_{N-1}

Según la fórmula de Newton, el polinomio de interpolación es:

$$p(t) = a_0 + a_1(t-t_0) + a_2(t-t_0)(t-t_1) + \dots + a_{N-1}(t-t_0)(t-t_1)\dots(t-t_{N-2})$$

Los coeficientes a(k) son las diferencias divididas. Para calcularlas de forma eficiente se construía lo que llamábamos el triángulo de diferencias divididas:



Simplificación al usar nodos equiespaciados

Este caso es especialmente sencillo al estar los nodos (t_k) equiespaciados:

$$t_k = t_0 + kh \qquad con \quad k = 0,1,\ldots,N-1$$

Donde h es la separación entre nodos. En nuestro caso h = 15 min = 900 s.

Podemos usar **diferencias finitas**, en vez de las diferencias divididas.

Más sencillas de calcular al no tener que dividir por la diferencia de tiempos.

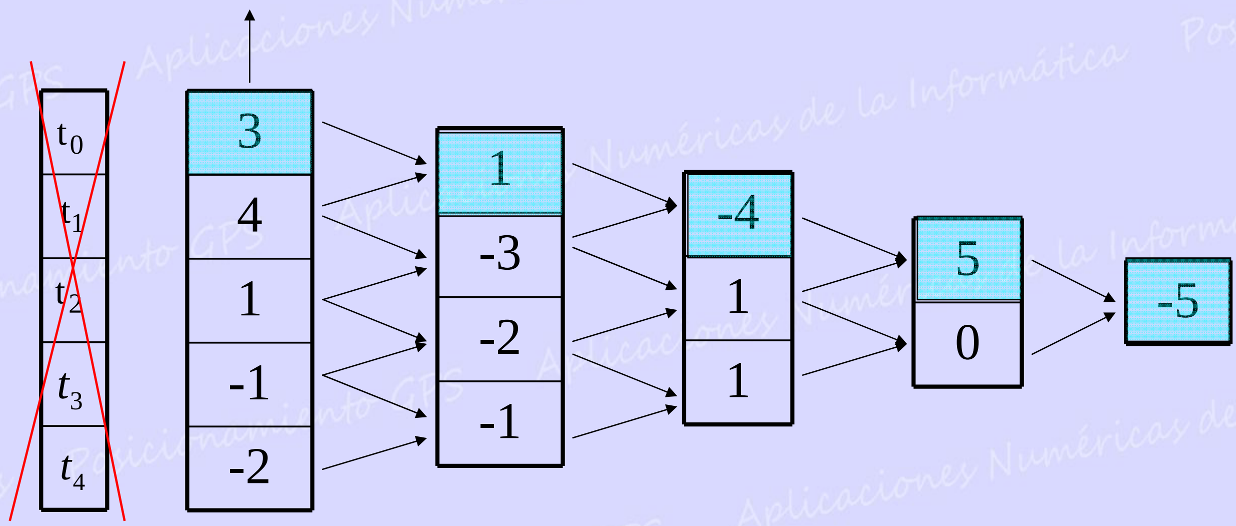
Simplemente hay que restar los términos vecinos de la columna anterior.

Nos ahorramos las divisiones y el código es más sencillo.

Al igual que antes, las diferencia finitas $\{d_k\}$ son los términos que quedan en la diagonal superior del triángulo.

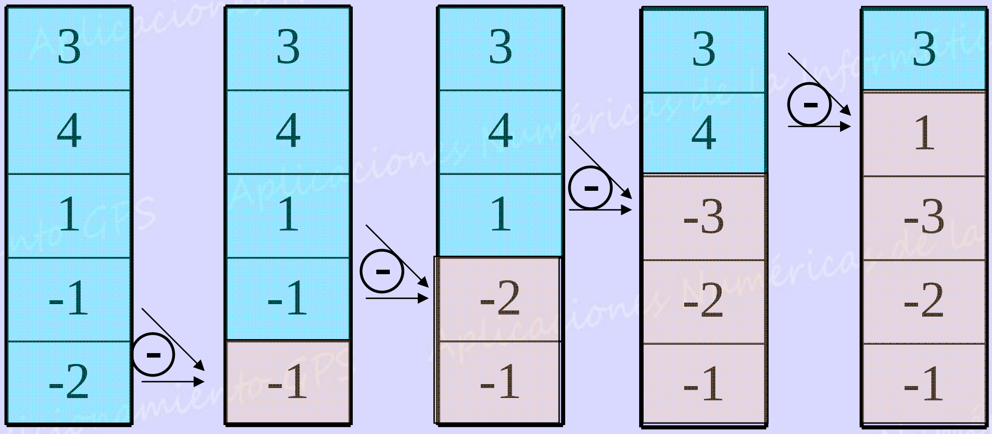
Ejemplo de tabla de diferencias finitas

Datos de partida = valores en los nodos = {3, 4, 1 , -1 ,-2}

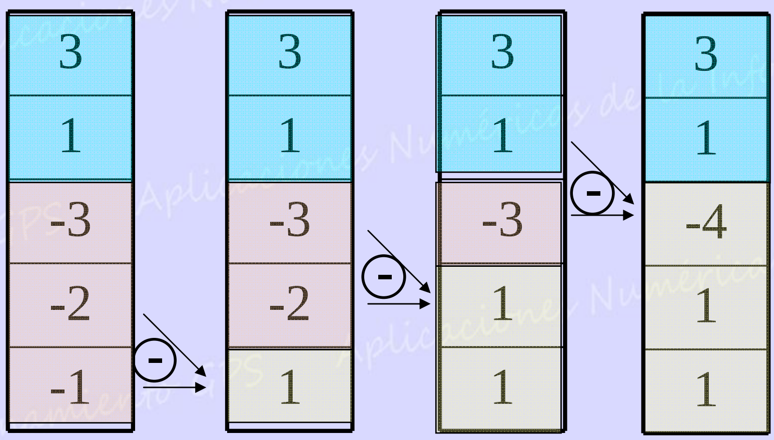


Diferencias finitas = {3, 1 , -4 , 5, -5}

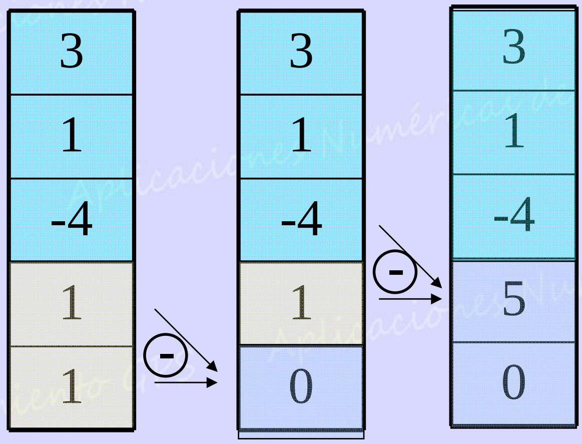
Cálculo sin necesidad de memoria adicional



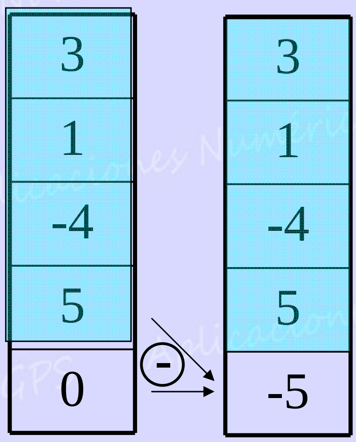
Cálculo sin necesidad de memoria adicional



Cálculo sin necesidad de memoria adicional



Cálculo sin necesidad de memoria adicional



Relación diferencias divididas / finitas.

Con puntos equi-espaciados no hace falta dividir porque la división es predecible:

1) En la primera columna de diferencias divididas se divide por la resta de {t_k} consecutivas. Al ser equi-espaciados esto siempre sale h: en la 1ª columna diferencias finitas y divididas difieren en un factor h.

2) En la 2ª columna las diferencias divididas se dividen por la resta de los {t_k} que ahora difieren en dos posiciones (2h). En esa columna la discrepancia acumulada entre diferencias divididas y finitas será ahora:

$$(\mathrm{h})\cdot(2\mathrm{h}) = 2\mathrm{h}^2$$

3) En la 3ª columna la división sería ahora por 3h, por lo que el factor total acumulado será:

$$(\mathrm{h})\cdot(2\mathrm{h})\cdot(3\mathrm{h}) = 6\mathrm{h}^3$$

...) Para la k-ésima columna, la relación entre las diferencias divididas {a_k} y las correspondientes diferencias finitas {d_k} es:

$$a_k = \frac{d_k}{k!\cdot h^k}$$

Fórmula Newton con diferencias finitas

$$p(t) = d_0 + \frac{d_1}{1\cdot h}(t-t_0) + \frac{d_2}{2!\cdot h^2}(t-t_0)(t-t_1) + \frac{d_3}{3!\cdot h^3}(t-t_0)(t-t_1)(t-t_2) + \dots$$

Hacemos un cambio de variable (t ←→ s): $\mathrm{t} = \mathrm{t}_0 + \mathrm{sh}$ $\quad s = \dfrac{t-t_0}{h}$

La nueva variable **s** corresponde a la **distancia de t respecto al 1er nodo de la tabla (t0)** medida en unidades de h (el espaciado de la tabla):

Usando la nueva variable: $(t-t_k) = \left(\cancel{t_0} + sh\right) - \left(\cancel{t_0} + kh\right) = (s-k)\cdot h$

y el polinomio de Newton en la nueva variable s queda como:

$$p(s) = d_0 + \frac{d_1}{1\cdot \cancel{h}}(s)\cdot \cancel{h} + \frac{d_2}{2!\cdot \cancel{h^2}}(s)(s-1)\cdot \cancel{h^2} + \frac{d_3}{3!\cdot \cancel{h^3}}(s)(s-1)(s-2)\cdot \cancel{h^3} + \dots$$

Simplificación de p(s) con diferencias finitas

$$p(s) = d_0 + d_1 \frac{(s)}{1} + d_2 \frac{(s)(s-1)}{2!} + d_3 \frac{(s)(s-1)(s-2)}{3!} + \ldots$$

$$p(s) = d_0 S_0 + d_1 S_1 + d_2 S_2 + \ldots + d_{N-1} S_{N-1} = \sum_{k=0}^{N-1} d_k S_k$$

El valor interpolado p(s) se calcula a partir de las N diferencias finitas $\{d_0, d_1, \ldots, d_{N-1}\}$ y N coeficientes $\{S_0, S_1, S_2, \ldots, S_{N-1}\}$:

a) Las diferencias finitas d_k dependen únicamente de los N valores de la tabla a interpolar (sin tener en cuenta sus tiempos).

b) Los coeficientes S_k solo dependen de la variable s, esto es, del tiempo t a interpolar y su relación con los nodos de la tabla, ya que:

$$\{S_k\} \longleftarrow s = \frac{(t - t_0)}{h}$$

t_0 → Tiempo del 1er nodo de la tabla

h → Espaciado entre nodos

Cálculo de los coeficientes S(k)

Los coeficientes S(k) puede construirse de forma iterativa a partir del anterior:

$$S_0 = 1$$

$$S_1 = 1 \cdot \frac{s}{1}$$

$$S_2 = \frac{s}{1} \frac{(s-1)}{2}$$

$$S_3 = \frac{s \cdot (s-1)}{1 \cdot 2} \cdot \frac{(s-2)}{3}$$

$$S_4 = \frac{s \cdot (s-1) \cdot (s-2)}{1 \cdot 2 \cdot 3} \frac{(s-3)}{4}$$

Dado $S_0=1$, $S_{k+1} = S_k \dfrac{(s-k)}{k+1}$

El coste es sólo de una multiplicación + división para calcular cada coeficiente.

Resumen del Proceso

1) Definir el número de nodos N a usar

2) A partir de t se calcula el valor de s. A partir de N y el tiempo t podemos calcular s como:

$$s = \frac{t - t_0}{h} = \left(\frac{N}{2} - 1\right) + \frac{\mathrm{mod}(t,h)}{h}$$

3) A partir de s se determinan los N coeficientes {Sk}. En MATLAB:

$S_1 = 1,$ for k = 1 hasta N-1, $S_{k+1} = \frac{(s-k+1)}{k} S_k$ end

4) Determinar a partir del tiempo t los índices de los N nodos a usar.

5) Para esos nodos, extraer los valores de X de la estructura SP3 y calcular sus diferencias divididas {dk}.

6) Conocidos {dk} y {Sk} calcular el valor interpolado de X como: $\sum_{k=1}^{N} S_k d_k$

7) Repetir 5) y 6) para los datos de las otras coordenadas Y, Z y cdT

Tarea para hacer ANTES del próximo día

Trabajando por parejas, completad las <u>dos funciones</u> que se piden en el primer ejercicio de la hoja de LABORATORIO.

1) DF = get_df(F): calcula las diferencias finitas DF a partir de un vector de datos F. Por ahora no hay que preocuparse de dónde salen esos datos.

2) S = get_S(s,N): calcular el vector de coeficientes {Sk} a partir del valor de s y el número de nodos N usados. Al igual que antes en esta función s es un dato de entrada, no nos preocupamos de cómo se calcula.

Bien escritas, ninguna de estas funciones debería ocupar más allá de unas 5/10 líneas de código.

No hay que entregar nada, pero estas funciones son necesarias para el objetivo del próximo día en el LAB: una rutina para obtener la posición + error de reloj de un satélite en cualquier instante de tiempo del día.

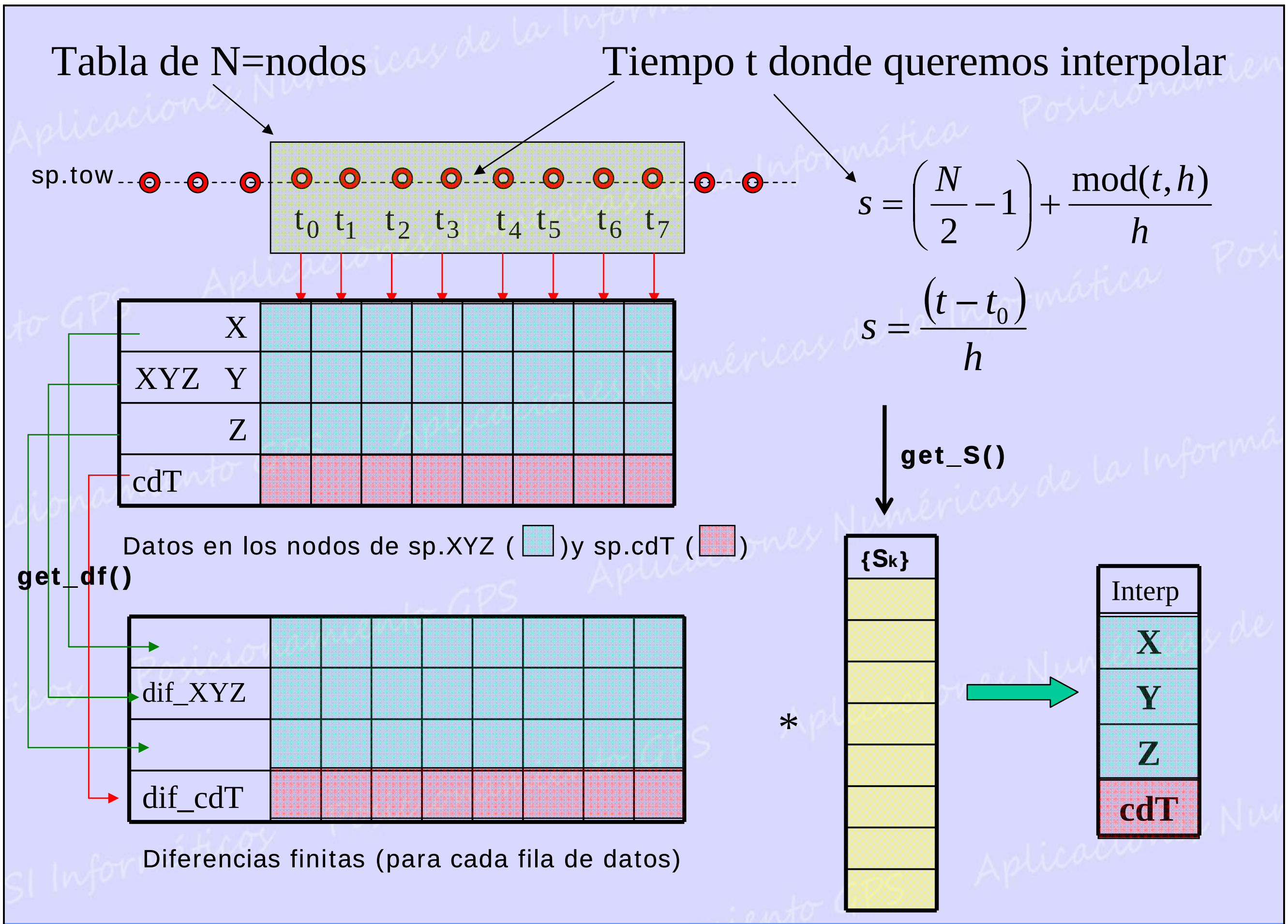


Tabla de N=nodos
Tiempo t donde queremos interpolar
sp.tow
t_0 t_1 t_2 t_3 t_4 t_5 t_6 t_7
$s = \left(\frac{N}{2} - 1\right) + \frac{\mathrm{mod}(t,h)}{h}$
$s = \frac{(t - t_0)}{h}$
X
XYZ Y
Z
cdT
get_S()
Datos en los nodos de sp.XYZ ( )y sp.cdT ( )
get_df()
{Sk}
Interp
dif_XYZ
*
X
Y
Z
dif_cdT
cdT
Diferencias finitas (para cada fila de datos)