REAL INSTITUTO Y OBSERVATORIO DE LA ARMADA EN SAN FERNANDO BOLETÍN ROA No. 2/2015



TRABAJOS FIN DE MÁSTER DE LOS ALUMNOS
DE LA ESCUELA DE ESTUDIOS
SUPERIORES DE LA ARMADA
CICLO 2014-15



MINISTERIO DE DEFENSA

TRABAJOS FIN DE MASTER DE LOS ALUMNOS DE LA ESCUELA DE ESTUDIOS SUPERIORES DE LA ARMADA

Sergio Borrallo Tirado

Lluís Canals Ros

José Ángel Cibeira Urtiaga



San Fernando, Junio 2015

CATÁLOGO GENERAL DE PUBLICACIONES OFICIALES

http://publicacionesoficiales.boe.es/

Edita:



© Autor y editor, 2015

NIPO: 083-15-221-0 (impresión bajo demanda)

NIPO: 083-15-222-6 (edición en línea)

ISSN: 1131-5040

Fecha edición: julio 2015

HISTORIUS COO

http://publicaciones.defensa.gob.es/

Las opiniones emitidas en esta publicación son de la exclusiva responsabilidad del autor de la misma.

Los derechos de explotación de esta obra están amparados por la Ley de la Propiedad Intelectual. Ninguna de las partes de la misma puede ser reproducida, almacenada ni transmitida en ninguna forma ni por medio alguno, electrónico, mecánico o de grabación, incluido fotocopias, o por cualquier otra forma, sin permiso previo, expreso y por escrito de los titulares del © *Copyright*.

Foto portada:

Fachada del Edificio Principal del Real Instituto y Observatorio de la Armada en San Fernando (siglo XVIII).

Relación de Trabajos presentados en este Boletín ROA

1. Nuevas variables canónicas para la integración analítica del problema principal-ecuatorial del satélite artificial

1.1. Autor : Sergio Borrallo Tirado

1.2. Páginas : 3 - 65

2. Automatic Reduction of GEO Survey Data

2.1. Autor : Lluís Canals Ros

2.2. Páginas : 66 – 112

3. Introducción al manejo del software GOA en PPP. Aplicación práctica a la geodinámica de la Isla de Alborán.

3.1. Autor : José Ángel Cibeira Urtiaga

3.2. Páginas : 113 - 191

Nuevas variables canónicas para

la integración analítica del

problema principal-ecuatorial del

satélite artificial

Sergio Borrallo Tirado

Máster en Modelización Matemática, Estadística y Computación Universidad de Zaragoza

Dedico esta memoria a mi esposa, Gökçe Yücel, porque sin su amor y apoyo nada es posible para mí.

Mis agradecimientos a mi tutor, D. Alberto Abad Medina, que ha tenido al paciencia y dedicación para orientarme en todo momento, y a los Oficiales profesores del Real Observatorio de la Armada, cuyas enseñanzas han sido de vital importancia para la finalización de esta memoria.

Nuevas variables canónicas para la integración analítica del problema principal-ecuatorial del satélite artificial

	Int	troducción 09-12	
1)) Formulación del problema principal-ecuatorial ·······		
	a)	Movimiento kepleriano	
	<i>b)</i>	Variables de estado: vector de estado	
	c)	Elementos orbitales	
	d)	Variables polares-nodales	
	e)	Variables de Delaunay	
	f)	Movimiento orbital	
	g)	Problema principal ecuatorial del satélite artificial	
2) Nuevas variables tipo "Delaunay" para la integración del problema		nevas variables tipo "Delaunay" para la integración l problema	
	a)	Obtención de la función generatriz	
	b) c)	Significado de los nuevos momentos Obtención de las nuevas variables	

	d)	Obtención de las funciones $\lambda(r)$ y $f(r)$
	e)	Significado de las nuevas variables
	f)	Expresiones explicitas de la transformación
	g)	Integración del problema principal-ecuatorial en las nuevas variables
3)	Ob	otención de la distancia r en función del tiempo t .
	Е	cuación de Kepler Generalizada 39-52
	a)	Expresión de la relación entre r y t como un oscilador armónico perturbado
	<i>b)</i>	Aplicación del método Krylov-Bogoliuvov-Mitropolsky
	c)	Relación de r con ψ y f
	d)	Ecuación de Kepler Generalizada
	Со	nclusiones 53-54
	Ap	éndices 55-62
	e)	Elementos de dinámica hamiltoniana y ecuación de Hamilton-Jacobi
	f)	Método de Krylov-Bogoliubov-Mitropolsky
	Bil	oliografía 63-65

Introducción

La tecnología espacial es responsable de una buena parte de los avances tecnológicos actuales. Desde el año 1957, cuando se lanzó al espacio el primer satélite artificial, se han realizado una gran variedad de misiones espaciales con objetivos muy diferentes. Una de las partes fundamentales de cada proyecto es el análisis de misión, que consiste en la determinación previa de la órbita que seguirá el satélite para cumplir las especificaciones de la misión (Werz&Larson, 2010). El poder imponer el cumplimiento de determinadas condiciones para una órbita ha permitido alcanzar con éxito muchos de los objetivos de las misiones realizadas hasta hoy.

De entre toda la variedad de órbitas de satélites artificiales terrestres las más conocidas, por su utilidad para las comunicaciones, son las llamadas órbitas geoestacionarias. Las primeras ideas sobre las órbitas geoestacionarias fueron publicadas por Herman Potočnik en 1928 en su trabajo "El problema del viaje espacial - El motor cohete". Años después, en 1945, Arthur C. Clarke hace mención a las ventajas que puede tener el uso de esta órbita para las comunicaciones, pues con solo tres satélites es suficiente para establecer comunicación con cualquier punto del planeta, exceptuando regiones cerca de los polos.

En el año 1963 es lanzado, y puesto en órbita, el Syncom-2,que es el primer satélite de comunicaciones en órbita geoestacionaria. Esto confirmó las ideas de Arthur C. Clarke y constituye un avance fundamental en distintas disciplinas o áreas de conocimiento como son: las telecomunicaciones, climatología, oceanografía, geodesia, militar, entre muchas otras más.

Las órbitas geoestacionarias pertenecen a un conjunto más amplio, que llamaremos *órbitas geosíncronas*, caracterizadas por su sincronía con la rotación de la Tierra, de manera que el periodo orbital del satélite, que es función del semieje mayor de la elipse orbital, coincide con el periodo de rotación terrestre, esto es, un día sidéreo (23h 56m 4.09s). Para que ésto sea así, el semieje (o el radio si son circulares) debe medir 42164 km. Si además de la sincronización de periodos hacemos que la inclinación de la órbita sobre el ecuador terrestre sea nula, esto es, que el satélite esté siempre en el ecuador, conseguimos que el satélite sea visible desde la Tierra como un punto fijo en el ecuador, de ahí que a estas órbitas se las llame *órbitas geoestacionarias*, y que una simple antena fija permita una recepción y transmisión de datos de forma continua.

Una detallada descripción de las características dinámicas de estas órbitas puede verse en Soop (1994). El objetivo de esta memoria es construir una teoría analítica,

basada en un conjunto especial de variables canónicas, para describir y analizar el movimiento de los satélites geoestacionarios o próximos a geoestacionarios.

La primera aproximación al movimiento orbital nos la dan las ecuaciones del movimiento kepleriano, que no son sino las ecuaciones derivadas de las leyes de Newton cuando existen fuerzas de atracción gravitacional entre masas puntuales. La integración de estas ecuaciones conduce a un movimiento que puede describirse a partir de las conocidas *leyes de Kepler*.

Las leyes de Kepler nos aseguran órbitas planas y cónicas y en ocasiones, cuando la cónica es una elipse, periódicas. Sin embargo, la realidad nos da órbitas muy próximas a las descritas por las leyes de Kepler pero no exactamente iguales, lo que se debe a la aparición de otras fuerzas, gravitacionales o no, que actúan sobre el sistema modificándolo. Entre las fuerzas más importantes que perturban la órbita de un satélite artificial tenemos: que la Tierra no es un punto sino un cuerpo no esférico; el rozamiento producido por la atmósfera cuando el satélite está dentro de ella; la fuerza gravitacional producida por el Sol y la Luna, la fuerza de radiación solar, etc. (Abad, 2012; Danby, 1988; Kovalevsky, 1967).

El *problema principal* del satélite artificial consiste en el estudio de su movimiento orbital considerando un modelo kepleriano perturbado por el achatamiento de la Tierra (o de cualquier otro planeta). El achatamiento de un planeta constituye la principal perturbación sobre la órbita kepleriana de un satélite artificial; cualquier otra perturbación tiene una magnitud mucho menor que ésta.

Las perturbaciones separan el satélite de su posición prevista, lo que obliga a encenderlos motores del satélite, cada cierto tiempo, para situarlo de nuevo en esa posición. Naturalmente ese gasto de combustible determina el coste y duración de la misión espacial, por lo que un buen conocimiento previo de la órbita que seguirá el satélite incide en su coste final.

Por ello resulta de gran utilidad la obtención de un modelo analítico del movimiento del satélite considerando el mayor número posible de perturbaciones que permita integrar el modelo.

A pesar de la simplicidad de un sistema dinámico kepleriano, la dinámica del movimiento orbital se complica notablemente cuando añadimos cualquier tipo de perturbación. Tanto si añadimos un tercer cuerpo al sistema orbital como si consideramos la perturbación producida por el achatamiento obtenemos sistemas dinámicos "no integrables" que nos conducen a la necesidad de aplicar métodos numéricos o desarrollos asintóticos para el análisis de los sistemas. La única posibilidad de obtener sistemas integrables consiste en restringir alguna de las características del movimiento, por ejemplo, si en el problema principal nos

limitamos a orbitas ecuatoriales, el problema, que llamaremos *problema principal-ecuatorial*, se convierte en integrable.

Por su integrabilidad consideraremos el problema principal-ecuatorial del satélite para trazar la órbita preliminar del satélite en sustitución de su órbita kepleriana. Aunque existen diversas soluciones analíticas a este problema (Jezewski 1983 y Martinusi&Gurfil 2013), ninguna de ellas permite su posterior utilización para el estudio del efecto de otras perturbaciones de menor magnitud, como el efecto gravitacional de la luna o la radiación solar, etc.

En este trabajo se propone un nuevo enfoque en la solución de este problema, basado en la aplicación de la ecuación de Hamilton-Jacobi para la obtención de un nuevo conjunto de variables canónicas, similares a las obtenidas por Delaunay para el movimiento de la Luna (*Delaunay 1860*), donde, al añadir la perturbación del problema principal-ecuatorial como un término μ^*/r^3 , se introducen expresiones dependientes de funciones e integrales elípticas que serán obtenidas a partir de las relaciones que aparecen en *Byrd* Friedman (1971).

Las variables de Delaunay, cuyo significado está intimamente relacionadas con los elementos orbitales, han jugado un papel fundamental en la obtención de las teorías analíticas del movimiento orbital. La primera de estas variables es la anomalía media, l que no es sino una función lineal del tiempo que varía de 0 a 2π en cada periodo, por lo que su uso equivale a usar explícitamente la variable tiempo en las expresiones de la teoría.

Una de las características del problema kepleriano es que resulta imposible la obtención de expresiones analíticas cerradas (que no usen desarrollo en serie de potencias) para ninguno de los parámetros dinámicos que describen el problema, en particular la distancia r entre el satélite y el planeta. Para resolver esto se usa, en lugar de la anomalía media, otra variable angular, llamada anomalía excéntrica, E, cuya relación con la anomalía media viene dada a partir de la llamada ecuación de Kepler, $l = E - e \sin E$, donde e es la excentricidad. Esta relación resulta de enorme importancia en el tratamiento del movimiento orbital.

En nuestro caso, aparecerá el mismo problema, esto es, no podremos expresar explícitamente ningún parámetro a partir de la primera de las variables obtenidas, por lo que deberemos desarrollar un procedimiento para encontrar la expresión equivalente a la ecuación de Kepler, que nosotros llamaremos ecuación de Kepler generalizada y que jugará, en este problema, un papel similar al de la ecuación de Kepler en el movimiento kepleriano.

La ecuación de Kepler generalizada se obtendrá de manera indirecta reformulando el problema principal-ecuatorial como un oscilador armónico perturbado que se resolverá aplicando el *método de Krylov-Bogoliuvov-Mitropolsky* (explicado en el apéndice B y que, a partir de ahora, será llamado por brevedad método KBM)

La presente memoria se presenta en tres capítulos que presentan respectivamente: la formulación del problema principal-ecuatorial del satélite; la obtención de unas variables similares a las de Delaunay adecuadas a este problema (junto con la integración del mismo); y la obtención de la ecuación de Kepler generalizada.

Finalmente, después de las conclusiones, se presentan varios apéndices con las herramientas básicas usadas a lo largo de la memoria, y la bibliografía utilizada.

1. Formulación del problema principal-ecuatorial.

Movimiento kepleriano

El **problema kepleriano** consiste en el estudio del movimiento de una partícula P, que llamaremos **orbitador¹**, relativo a otra partícula O, que llamaremos **cuerpo central²**. Las ecuaciones del movimiento kepleriano están representadas por el siguiente sistema de ecuaciones diferenciales:

$$\ddot{\boldsymbol{x}} = \boldsymbol{F}_k = -\frac{\mu}{r^3} \boldsymbol{x},$$

donde $F_k(x,\mu)$ es la fuerza kepleriana, siendo $x = \overline{OP}$ el vector de posición de P respecto de O, r = ||x|| la distancia y $\mu = \mathcal{G}(m_O + m_P)$ la constante gravitatoria resultado del producto de la constante de gravitación universal \mathcal{G} por la suma de las masas de ambos cuerpos.

El sistema diferencial anterior, de orden dos, puede ponerse como un sistema de ecuaciones de orden uno en la forma

$$\dot{\mathbf{x}} = \mathbf{X},\tag{1.1}$$

$$\dot{\mathbf{X}} = -\frac{\mu}{r^3}\mathbf{X},$$

donde**X** es el vector velocidad.

El problema kepleriano puede ser también expresado en forma hamiltoniana. Para ello supondremos un sistema dinámico cuyo hamiltoniano, que llamaremos *hamiltonianokepleriano*, tendrá la forma

$$\mathcal{H}_k(\mathbf{x}, \mathbf{X}) = T + V_k = \frac{1}{2}\mathbf{X} \cdot \mathbf{X} - \frac{\mu}{\|\mathbf{x}\|},$$

donde x representa las coordenadas y Xlos momentos y hemos llamado energía cinética del movimiento relativo al primer sumando y energía potencial del movimiento relativo al segundo.

-

¹ Entre los distintos tipos de orbitadores encontramos, entre otros, planetas, asteroides, cometas, satélites naturales y artificiales, etc.

² Estrellas como el Sol, planetas, etc.

Las ecuaciones de Hamilton correspondientes al hamiltoniano \mathcal{H}_k son

$$\dot{x}=
abla_X\mathcal{H}_k=X,$$
 $\dot{X}=-
abla_x\mathcal{H}_k=-rac{\mu}{r^3}x$,

y por tanto son idénticas a las ecuaciones (1.1) del movimiento kepleriano, por ello, podemos concluir que éste está representado por un sistema dinámico de hamiltoniano \mathcal{H}_k .

Variables de estado: vector de estado

Llamaremos *órbita kepleriana* y la denotaremos con el símbolo \mathcal{O} , a la solución de las ecuaciones del problema kepleriano para unas condiciones iniciales dadas. Entenderemos por órbita no solo la trayectoria del orbitador, sino todos sus parámetros, tanto estáticos-constantes como dinámicos-variables.

Las ecuaciones del problema kepleriano constituyen un sistema de seis ecuaciones diferenciales de orden uno. De acuerdo con la teoría de ecuaciones diferenciales ordinarias la solución general de dicho sistema vendrá dada como $\mathbf{x} = \mathbf{x}(t, \mathbf{C})$, donde $\mathbf{C} = (C_1, C_2, C_3, C_4, C_5, C_6)$ representa un vector de seis constantes funcionalmente independientes que llamaremos **variables de estado** porque permiten determinar cualquier parámetro de la órbita en cualquier instante, es decir, caracterizan la órbita.

Los seis elementos que componen las variables de estado son constantes de la órbita o variables dinámicas particularizadas para un instante dado. En este último caso hay que dar el valor de éstas y el instante t_0 en que han sido calculadas.

Una vez determinado el conjunto de variables de estado, la órbita quedará caracterizada por éste, y pondremos $\mathcal{O}(\mathbf{C})$ si los elementos del vector de estado son constantes de la órbita y $\mathcal{O}(t_0, \mathbf{C})$ si son variables particularizadas en t_0 .

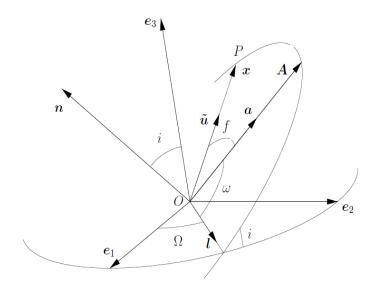
Las variables de estado pueden ser elegidas de diversas maneras. La más natural, desde el punto de vista de las ecuaciones diferenciales, es a través de los valores del vector de posición, x_0 , y la velocidad, X_0 , para un instante dado. El vector de dimensión seis formado por las componentes de los vectores x_0 y X_0 se llama vector de estado. De esta forma una órbita kepleriana podrá ser representada como $\mathcal{O}(t_0, x_0, X_0)$.

El vector de estado es la forma más natural de representar una órbita desde el punto de vista de las ecuaciones diferenciales que rigen la evolución del sistema, sin embargo existen otros conjuntos de variables de estado que se adaptan mejor a diferentes aspectos del movimiento kepleriano.

Por un lado los elementos orbitales constituyen la forma más natural, desde el punto de vista geométrico y astronómico, de visualizar y comprender como son las órbitas de los astros. Por otro lado, existen otros conjuntos de variables de estado, como son las variables polares-nodales y las variables de Delaunay que por su carácter de variables canónicas permiten el estudio del movimiento con técnicas de dinámica hamiltoniana.

Elementos orbitales

Para representar una órbita de forma mucho más descriptiva, tanto desde un punto de vista geométrico como astronómico y dinámico, se utilizan unas variables de estado que se adaptan completamente a la descripción del movimiento dada por las leyes de Kepler: los *elementos orbitales*.



Órbita kepleriana en el espacio

En primer lugar tomaremos los dos elementos que caracterizan la cónica descrita por un cuerpo que sigue las leves de Kepler, esto el semieje mayor (o semi-lado recto p) y la excentricidad e, representan respectivamente las dimensiones y la forma de la cónica.

Para completar la información sobre la trayectoria necesitaremos situarla en el espacio, para lo cual basta observar la figura que se muestra arriba y recordar que la órbita está contenida en un plano perpendicular al vector momento angular $G = x \times X$ o, lo que es igual, a su dirección n. Supondremos, por ahora, que la órbita no coincide con el plano Oxy del sistema espacial, esto es, que $n \times e_3 \neq 0$.

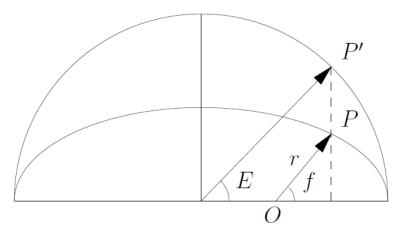
Puesto que el plano de la órbita y el plano fundamental del sistema espacial Oxy no son paralelos, necesariamente se cortarán en una recta que pasa por O y pertenece a ambos planos y que llamaremos *línea de los nodos*. Tomaremos como dirección positiva de dicha recta la que contiene el nodo ascendente, o punto de la órbita en el que el orbitador pasa de coordenadas z negativas a positivas. El vector unitario l define la línea de los nodos y forma un ángulo Ω , llamado ángulo del nodo, con e_1 . El ángulo Ω puede tomar cualquier valor entre 0 y 2π .

El ángulo que forman el vector \mathbf{n} con $\mathbf{e_3}$ será llamado *inclinación*, y denotado por i, y representa también el ángulo entre el plano 0xy y el de la órbita. El ángulo i puede tomar un valor cualquiera entre 0 y π .

El vector \boldsymbol{n} representa también el sentido de la rotación de la partícula pues, debido a su definición, ésta tiene siempre lugar en sentido contrario a las agujas del reloj si se observa desde el extremo de \boldsymbol{n} . De acuerdo con esto, el ángulo \boldsymbol{i} que forma \boldsymbol{n} con $\boldsymbol{e_3}$ indica el sentido de giro observado desde un punto cualquiera de la parte positiva del eje Oz. Un ángulo \boldsymbol{i} entre O y $\frac{\pi}{2}$ indicará una órbita directa (sentido de giro contrario a las agujas del reloj), mientras que una inclinación entre $\frac{\pi}{2}$ y π indicará una órbita retrógrada (sentido de giro igual al de las agujas del reloj).

Los dos ángulos, Ω e i representan la posición del plano de la órbita en el espacio, pero para poder representar con exactitud la situación de la cónica hay que determinar la dirección del eje de la misma dentro de su plano. El eje de la cónica, también llamado *línea de los ápsides*, lleva la dirección de un vector a, que forma un ángulo w con la línea de los nodos. Dicho ángulo será llamado *argumento del periastro*, representa la posición relativa de la cónica en su plano y es la tercera variable angular de la órbita. El argumento del periastro toma un valor cualquiera entre 0 y 2π .

Se han completado así los cinco elementos que caracterizan la geometría de la órbita, esto es, la forma, dimensiones y situación de la curva que recorre el orbitador. Para completar la caracterización de la órbita bastará un elemento que describa la dinámica, o lo que es igual, que nos informe acerca del punto de la curva o trayectoria en el que se encuentra el orbitador en cada instante t.



Anomalías en el movimiento kepleriano

Para ello recordemos que si se elige un sistema de coordenadas polares (r, f), en el plano de la órbita (ver figura anterior), con origen en el cuerpo central 0, y con eje de coordenadas polares según la dirección del periastro, entonces la ecuación de una cónica nos da la relación

$$r = \frac{p}{1 + e \cos f} \; ,$$

donde f es llamada anomalía verdadera.

Para encontrar la relación de f con t se introducen dos nuevas variables angulares: la anomalía excéntricaE y la anomalía medial = n(t-T), siendo $n=\sqrt{\mu/\alpha^3}$, el movimiento medio, y T la época o instante de paso del orbitador por el periastro. Las relaciones entre las tres anomalías vienen dadas por la expresión

$$\tan\frac{f}{2} = \sqrt{\frac{1+e}{1-e}} \tan\frac{E}{2} ,$$

que permite obtener fácilmente f en función de E, y E en función de f, y por otro lado por

$$l = E - e \operatorname{sen} E, \tag{1.2}$$

llamada ecuación de Kepler, que juega un papel fundamental en la Astrodinámica, y que nos da la relación de l con E, y por tanto con f. Así como la relación anterior ente f y E es fácilmente invertible, la ecuación de Kepler no lo es y permite obtener fácilmente l conocido E, sin embargo, para obtener E a partir de l, debemos aplicar un método numérico u obtener la inversa a partir de un desarrollo en serie de potencias de la excentricidad.

Para caracterizar su dinámica basta considerar finalmente la constante ${}^{3}T$, puesto que a partir de ella y las relaciones entre las anomalías puede obtenerse la posición del orbitador en su trayectoria.

Llamaremos *elementos orbitales* al conjunto de seis constantes (a, e, i, Ω, w, T) . La obtención de los elementos orbitales a partir de las condiciones iniciales (t_0, x_0, X_0) y viceversa, cuya demostración puede verse en *Abad (2012)*, demuestra la equivalencia entre ambos, y en consecuencia los elementos orbitales constituyen un conjunto de variables de estado. Por tanto, la órbita kepleriana se puede caracterizar como $\mathcal{O}(a, e, i, \Omega, w, T)$ o bien como $\mathcal{O}(t_0, x_0, X_0)$.

Variables polares-nodales

Se puede definir otro conjunto de variables que nos servirán igualmente para caracterizar una órbita dada. Al conjunto de variables $(r, \theta, \nu, R, \Theta, N)$ se le llama **variables polares-nodales** y en otras ocasiones variables de Hill o variables de Whitaker y constituyen un conjunto de variables canónicas donde (r, θ, ν) son las variables y (R, Θ, N) los momentos.

La relación de estas variables con los elementos orbitales viene dada por las expresiones

$$r$$
 ; $\theta=w+f$; $\nu=\Omega$, $R=\dot{r}$; $\Theta=G$; $N=G(e_3\cdot n)=G\cos i.$

Notemos que R representa la velocidad radial y N la proyección del momento angular sobre el eje OZ. A la variable θ se le llama argumento de latitud⁴.

Al constituir las variables polares-nodales un conjunto de variables canónicas, podremos expresar el hamiltoniano del modelo kepleriano en estas variables, para lo cual basta tener en cuenta

$$X^2 = \dot{r}^2 + r^2 \dot{f}^2 = R^2 + \frac{\Theta^2}{r^2}$$
,

por lo que la función de Hamilton del problema kepleriano podrá ponerse como

$$\mathcal{H}_k = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) - \frac{\mu}{r} . \tag{1.3}$$

³ Aunque el elemento T es constante hay que tener en cuenta que, para órbitas elípticas, éste varía de una vuelta a otra aumentando en una cantidad igual al periodo orbital P.

⁴ Ángulo comprendido entre las rectas que unen el centro de masas del cuerpo principal con el nodo ascendente y el periastro de la órbita de un satélite.

Variables de Delaunay

Otro conjunto de variables muy útil e importante en el estudio del movimiento orbital son las variables de Delaunay (Delaunay, 1860) introducidas por éste para el estudio del movimiento de la Luna.

Las *variables de Delaunay*(l, g, h, L, G, H), constituyen también un conjunto de variables canónicas, donde (l, g, h), representan las variables y (L, G, H), los momentos, y que se definen por las expresiones

$$l=n(t-T)$$
 ; $g=w$; $h=\Omega$,
 $L=\sqrt{\mu\,a}$; G ; $H=G\cos i$,

donde puede destacarse que la primera variable representa la anomalía media, o lo que es igual el tiempo multiplicado por una constante.

Al igual que en el caso de las variables polares-nodales, las variables de Delaunay son canónicas, por lo que podremos representar⁵ el hamiltoniano del problema en estas variables en la forma

$$\mathcal{H}_k = -\frac{\mu^2}{2L^2} \ .$$

Movimiento Orbital

En la realidad, además de la fuerza kepleriana aparecen una serie de perturbaciones que modifican las ecuaciones y la trayectoria de la órbita, entre las que podemos destacar:

- ➤ Potencial gravitatorio del cuerpo central debido a la distribución de masa no esférica de los cuerpos celestes.
- Atracción gravitatoria de otros cuerpos del sistema distintos del central: la Luna y el Sol en el caso de órbitas terrestres, y la Tierra para órbitas lunares, etc.
- Rozamiento o resistencia de la atmosfera en aquellos cuerpos que la poseen.
- Presión de radiación solar.

⁵La demostración de esto puede verse en Abad (2012) y es similar a la que aparece para las variables de tipo Delaunay en el apartado segundo de esta memoria.

Efectos de mareas, relatividad y otros.

Las anteriores perturbaciones pueden formularse por medio de una fuerza adicional \mathcal{P} que modifica las ecuaciones del movimiento(1.1) en la forma:

$$\ddot{x} + \mu \frac{x}{r^3} = \mathcal{P},\tag{1.4}$$

o expresado como un sistema de orden uno

$$\dot{x} = X,$$

$$\dot{X} = -\frac{\mu}{r^3}x + \mathcal{P}.$$
(1.5)

Cuando el vector perturbación vale cero, $\mathcal{P}=0$, el movimiento coincide con el movimiento kepleriano, mientras que cuando se verifique la condición $\|\mathcal{P}\| \ll \frac{\mu}{r^2}$, esto es, cuando la aceleración que produce la perturbación sea mucho menor que la kepleriana, la solución de (1.4) o (1.5) será llamada *movimiento kepleriano perturbado* o, simplemente, *movimiento orbital*. En este caso el comportamiento del movimiento orbital es muy próximo al del movimiento kepleriano, sirviendo éste como primera aproximación para comprender y estudiar el movimiento de los cuerpos en el espacio. De hecho, la forma práctica para visualizar este movimiento orbital supone considerar que la órbita es instantáneamente kepleriana, esto es que en cada instante se puede describir como una partícula describiendo una cónica (órbita osculatriz) cuyos elementos van variando lentamente con el tiempo.

Si existe una función V_p que verifique

$$\mathbf{\mathcal{P}}=-\nabla_{x}V_{p}$$
 ,

podemos definir el hamiltoniano \mathcal{H} , del movimiento orbital

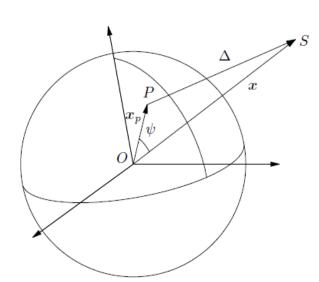
$$\mathcal{H}(x,X) = \mathcal{H}_k + V_p = \frac{1}{2}X \cdot X - \frac{\mu}{\|x\|} + V_p,$$

como suma del hamiltonianokepleriano \mathcal{H}_k y la función V_p , que llamaremos potencial perturbador.

Las ecuaciones de Hamilton aplicadas a este hamiltoniano coinciden con las ecuaciones (1.5) del movimiento orbital, por lo que ambos sistemas son equivalentes y llamaremos a $\mathcal{H}(x,X)$ hamiltoniano del movimiento orbital.

Problema principal ecuatorial del satélite artificial

Las ecuaciones diferenciales del modelo kepleriano están basadas en la ley de gravitación universal de Newton que formula la fuerza de atracción entre dos masas puntuales. En un caso más general dicha fuerza de atracción puede extenderse a sólidos homogéneos de forma esférica. Sin embargo en el caso del satélite artificial los dos cuerpos involucrados son el satélite y la Tierra (o planeta). La pequeña masa y dimensiones del satélite comparada con la de la Tierra permite considerar éste como un punto, sin embargo la forma de la Tierra no es esférica, de hecho, ni siquiera es un sólido, sino un fluido con movimientos internos de masas y una densidad no constante, a pesar de lo cual consideraremos como aproximación suficiente una Tierra o planeta sólido de densidad constante. En lo que sigue formularemos el modelo de potencial gravitatorio producido por un sólido homogéneo sobre una masa puntual.



Tal y como se ve en la figura de la izquierda, cada punto P de un sólido, de masa dm, ejerce sobre un orbitador S una fuerza de atracción cuyo potencial viene dado por la expresión $-\mathcal{G} \cdot \frac{dm}{\Delta}$ siendo Δ la distancia entre P y S, y \mathcal{G} la constante de gravitación universal. Además tomaremos como unidad de masa la de S y llamaremos dm al elemento diferencial de masa del

El potencial creado por el sólido en S vendrá dado por la integral extendida a toda la masa del solido

$$V = -\mathcal{G} \int_{M} \frac{dm}{\Delta} \,, \tag{1.6}$$

Si x, x_p representan los vectores de posición respectivos de S y P, referidos a un sistema con centro en el centro de masas del sólido, y ψ el ángulo entre dichos vectores, tendremos

$$\Delta^{2} = (x - x_{p})^{2} = x^{2} + x_{p}^{2} - 2|x||x_{p}|\cos\psi = |x|^{2}(1 - 2x\alpha + x^{2}).$$

Donde en la última expresión hemos llamado $x = \frac{|x_p|}{|x|}$, $\alpha = \cos \psi$. Finalmente, puesto que r = |x|, podemos poner

$$\frac{1}{\Delta} = \frac{1}{r} \cdot \frac{1}{\sqrt{1 - 2x\alpha + x^2}} \,. \tag{1.7}$$

El término $1/\sqrt{1-2x\alpha+x^2}$ suele sustituirse por su desarrollo en serie de potencias

$$\frac{1}{\sqrt{1-2x\alpha+x^2}} = \sum_{n\geq 0} P_n(\alpha)x^n = \sum_{n\geq 0} P_n(\cos\psi)x^n,$$

cuyos coeficientes son los polinomios de Legrende, definidos por la expresión

$$P_n(t) = \frac{1}{2^n n!} \frac{d^n}{dt^n} (t^2 - 1)^n,$$

que en particular para, el orden dos, se puede poner como

$$P_2(t) = \frac{1}{2}(3t^2 - 1). \tag{1.8}$$

Para calcular la integral (1.6) a lo largo de toda la masa del solido formularemos el problema en un sistema de coordenadas planetográficas⁶ donde llamaremos (λ, ϕ) , respectivamente a la longitud y latitud planetográfica del satélite y (Λ, Φ) a las de un punto P del planeta y de igual modo vamos a llamar $\widehat{x_s}$ y $\widehat{x_p}$ a las direcciones de los vectores de posición, en coordenadas cartesianas, del planeta y del satélite respectivamente.

El coseno del ángulo ψ entre estos dos vectores vendrá dado por el producto escalar de ambos, lo que lleva a la expresión

$$cos\psi = \widehat{x_s} \cdot \widehat{x_p} = sen\phi \ sen\Phi + cos\phi \ cos\Phi \cos(\Lambda - \lambda).$$

La relación anterior, llevada a los polinomios de Legrende, permite obtener la siguiente propiedad

$$P_{n}(\cos\psi) = P_{n}(\operatorname{sen}\phi)P_{n}(\operatorname{sen}\Phi)$$

$$+ 2\sum_{j=1}^{n} \frac{(n-j)!}{(n+j)!} P_{nj}(\operatorname{sen}\phi)P_{nj}(\operatorname{sen}\Phi)\cos j(\Lambda - \lambda).$$
(1.9)

⁶ La longitud planetográfica se mide en el ecuador del planeta desde un meridiano principal definido y aceptado internacionalmente. La latitud planetográfica se mide en el arco que une el ecuador con la posición del punto en cuestión.

donde P_{nj} representan los polinomios asociados de Legendre (Abad, 2012) definidos a partir de las derivadas de los polinomios de Legendre.

Por otro lado, la expresión (1.6) del potencial del sólido, haciendo uso de (1.7), se podrá poner como

$$V = -\mathcal{G} \int_{M} \frac{dm}{\Delta} = -\mathcal{G} \int_{M} \frac{1}{r} \cdot \frac{1}{\sqrt{1 - 2x\alpha + x^{2}}} dm$$
$$= -\mathcal{G} \int_{M} \frac{1}{r} \cdot \sum_{n \ge 0} P_{n}(\cos\psi) x^{n} dm = -\frac{\mathcal{G}}{r} \sum_{n \ge 0} \int_{M} \left(\frac{\rho}{r}\right)^{n} P_{n}(\cos\psi),$$

de donde, usando la relación(1.9) llegamos a

$$V = -\frac{GM}{r} + \frac{GM}{r} \sum_{n \ge 2} \left(\frac{r_p}{r}\right)^n \left[J_n P_n(sen\phi) - \sum_{j=1}^n P_{nj}(sen\phi) \left(C_{nj} \cos j\lambda + S_{nj} \sin j\lambda \right) \right], \qquad (1.10)$$

siendo r_p el radio ecuatorial del planeta, M la masa del mismo y donde J_n , C_{nj} , S_{nj} , llamadas *coeficientes armónicos*⁷, son constantes para cada sólido que corresponden a las expresiones de las integrales después de separar los términos que dependen únicamente de la posición del orbitador. Si conocemos la distribución de masas del sólido estas constantes pueden ser calculadas por integración, sin embargo, en el caso de los planetas y la Tierra dichas constantes únicamente pueden ser medidas en forma indirecta a partir de un detallado análisis de las órbitas de los satélites artificiales.

Los términos J_n son llamados **armónicos zonales**, mientras que C_{nj} , S_{nj} son los **armónicos teserales**. Si el sólido tiene simetría de revolución, los términos dependientes de la longitud λ desaparecen, y nos quedamos únicamente con los términos zonales J_n . En particular el término J_2 corresponde al achatamiento planetario, esto es a la diferencia entre los radios ecuatorial y polar.

Por lo que respecta a su magnitud el armónico J_2 es en general mucho más grande que el resto de los armónicos, en el caso de la Tierra éste tiene una magnitud de 10^{-3} , mientras que el resto de armónicos es del orden de 10^{-6} . Esto significa que la expresión (1.10)puede separarse en tres sumandos. El primer sumando corresponde al término

Ī

⁷Eligiendo el centro de masas del sólido como origen del sistema de referencia los términos J_1 , C_{11} , $S_{11} = 0$, por lo cual el sumatorio (1.9) comienza en el índice n = 2 y no en n=1.

$$V_k = -\frac{\mathcal{G}M}{r},$$

que representa el potencial del planeta si fuese un punto o un sólido homogéneo, esto es el potencial kepleriano. El segundo sumando

$$V_p = \frac{GM}{r} \left(\frac{r_p}{r}\right)^2 J_2 P_2(sen\phi),$$

corresponde a la acción del achatamiento terrestre o lo que es igual a considerar el planeta como un elipsoide homogéneo de revolución. Finalmente, el tercer sumando corresponde a la parte de la expresión (1.10) excluyendo los sumandos V_k y V_p .

Como se ha dicho en el apartado anterior la primera aproximación al movimiento orbital del satélite artificial se obtiene utilizando el modelo kepleriano, lo que equivale a considerar $V = V_k$. Si queremos una mejor aproximación a la órbita real debemos añadir a éste el potencial debido al achatamiento terrestre, esto es, considerar el potencial

$$V = V_k + V_p = -\frac{\mu}{r} + \frac{\mu}{r} \left(\frac{r_p}{r}\right)^2 J_2 P_2(sen\phi), \tag{1.11}$$

siendo $\mu = \mathcal{G}M$. Al modelo obtenido sustituyendo el potencial kepleriano por (1.11) se le denomina *problema principal del satélite*.

En nuestro caso, como se ha dicho en la introducción, no consideraremos este modelo, sino una aproximación al mismo, que llamaremos *problema principal-ecuatorial* que consiste en suponer que el satélite se encuentra en una órbita que no abandona el plano ecuatorial, por lo que en todo momento $\phi = 0$, o lo que es igual, $sen\phi = 0$. De esta forma se tendrá, de acuerdo con (1.9), que $P_2(sen\phi) = -1/2$, y por tanto el potencial para el problema principal ecuatorial se expresará, a partir de (1.11) como

$$V = V_k + V_{pe} = -\frac{\mu}{r} + \frac{\mu^*}{r^3},$$
 (1.12)

donde $\mu = \mathcal{G}M \text{ y } \mu^* = -\mathcal{G}Mr_p^2J_2/2.$

Una vez conocido el potencial podemos obtener fácilmente la expresión del Hamiltoniano de este problema que, de acuerdo con (1.3), puede expresarse en variables polares-nodales como

$$\mathcal{H} = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) - \frac{\mu}{r} + \frac{\mu^*}{r^3} . (\mathbf{1}. \, \mathbf{13})$$

Si tenemos en cuenta que el factor J_2 es del orden de la milésima podemos concluir que $\mu^* \ll \mu$, y por tanto poner $\mu^* = \varepsilon \sigma$, donde ε es un pequeño parámetro y σ es del mismo orden que μ . De este modo, podemos expresar (1.13)en la forma

$$\mathcal{H} = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) - \frac{\mu}{r} + \varepsilon \frac{\sigma}{r^3} , \qquad (1.14)$$

esto es, como un desarrollo en serie de potencias de un pequeño parámetro ε , cuyo orden cero corresponde con el hamiltoniano kepleriano y el orden uno, σ/r^3 , representa la perturbación del problema principal ecuatorial.

2. Nuevas variables tipo "Delaunay" para la integración del problema

Para encontrar unas nuevas variables canónicas adecuadas para el problema principal-ecuatorial partiremos del hamiltoniano del mismo expresado en variables polares-nodales, que como hemos visto en el apartado anterior puede expresarse como

$$\mathcal{H}(r,\theta,v,R,\Theta,N) = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) - \frac{\mu}{r} + \frac{\mu^*}{r^3}, \qquad (2.1)$$

donde, como vemos, no aparecen explícitamente ni la variable v ni el momento N.

La no aparición de v (ángulo del nodo) y N (proyección del momento angular sobre el eje OZ) permite dejar fijos estos elementos en la transformación canónica que buscamos, de manera que trataremos de encontrar una transformación

$$(r, \theta, v, R, \Theta, N) \leftrightarrow (\lambda, \gamma, v, \Lambda, \Gamma, N),$$

que transforme únicamente las variables (r, θ) en (λ, γ) , y los momentos en (R, Θ) en (Λ, Γ) .

El procedimiento que seguiremos para obtener esta transformación es similar al usado en *Abad (2012)* para la obtención de las variables de Delaunay con la diferencia de que allí se parte del Hamiltoniano del problema kepleriano y aquí se parte del hamiltoniano (2.1) cuyo término adicional, μ^*/r^3 , complica las expresiones de las nuevas variables al aparecer integrales elípticas en las expresiones. En *Abad (1985)* se aplica un proceso similar para el estudio del movimiento de una estrella en un sistema triple.

Este procedimiento está basado en la búsqueda de una función generatriz $S(r, \theta, \Lambda, \Gamma)$, dependiente de las variables viejas y los nuevos momentos, que cumpla la ecuación de Hamilton-Jacobi, en la forma que se muestra en el apéndice A. Las expresiones (A.4)del apéndice indican que la transformación canónica se obtendrá a partir de las relaciones

$$R = \frac{\partial S}{\partial r}$$
, $\Theta = \frac{\partial S}{\partial \theta}$, $\lambda = \frac{\partial S}{\partial \Lambda}$, $\gamma = \frac{\partial S}{\partial \Gamma}$, (2.2)

mientras que la función generatriz, de acuerdo con (A.5), debe verificar

$$\frac{1}{2} \left[\left(\frac{\partial S}{\partial r} \right)^2 + \frac{1}{r^2} \left(\frac{\partial S}{\partial \theta} \right)^2 \right] - \frac{\mu}{r} + \frac{\mu^*}{r^3} = \Lambda , \qquad (2.3)$$

donde como se ve se ha elegido la energía del sistema como el primer momento Λ .

Obtención de la función generatriz

Para encontrar una función generatriz que verifique (2.3) ensayaremos una función

$$S(r, \theta, \Lambda, \Gamma) = S_1(r, \Lambda, \Gamma) + S_2(\theta, \Lambda, \Gamma),$$

que sea separable en otras dos, cada una de las cuales depende únicamente de una de las variables antiguas. De esta forma podremos poner

$$\frac{\partial S}{\partial r} = \frac{\partial S_1}{\partial r}$$
 ; $\frac{\partial S}{\partial \theta} = \frac{\partial S_2}{\partial \theta}$,

por lo cual, la ecuación (2.3) toma la forma:

$$\frac{1}{2} \left[\left(\frac{\partial S_1}{\partial r} \right)^2 + \frac{1}{r^2} \left(\frac{\partial S_2}{\partial \theta} \right)^2 \right] - \frac{\mu}{r} + \frac{\mu^*}{r^3} = \Lambda,$$

que, reorganizando términos, se transforma en:

$$\left(\frac{\partial S_2}{\partial \theta}\right)^2 = 2r^2 \Lambda - r^2 \left(\frac{\partial S_1}{\partial r}\right)^2 + 2\mu r - \frac{2\mu^*}{r},$$

donde el término de la izquierda solo depende de los nuevos momentos y de θ , mientras que el de la derecha solo depende de los momentos y de r. Así pues esto obliga a que ambos términos sean independientes de las variables r y θ , por lo que cualquier función de Λ y Γ verificará la igualdad. Por ello elegimos el momento Γ^2 como el valor de la igualdad, de forma que se tendrá

$$\left(\frac{\partial S_2}{\partial \theta}\right)^2 = 2r^2 \Lambda - r^2 \left(\frac{\partial S_1}{\partial r}\right)^2 + 2\mu r - \frac{2\mu^*}{r} = \Gamma^2. \quad (2.4)$$

Igualando el primer y el tercer término de la igualdad (2.4) se tiene que

$$\left(\frac{\partial S_2}{\partial \theta}\right)^2 = \Gamma^2 \quad \rightarrow \quad \frac{\partial S_2}{\partial \theta} = \pm \Gamma \quad \rightarrow \quad S_2(\theta) = \pm \Gamma \theta,$$

donde hemos elegido un límite inferior $\theta=0$ para la integración. Aunque cualquiera de las dos soluciones es válida elegiremos, para simplificar, la expresión con signo positivo, así se tendrá

$$S_2(\theta, \Lambda, \Gamma) = \Gamma \theta$$
.

De igual modo usando el primer y segundo término de (2.4) observamos que

$$2r^2\Lambda - r^2\left(\frac{\partial S_1}{\partial r}\right)^2 + 2\mu r - \frac{2\mu^*}{r} = \Gamma^2,$$

que reordenando queda:

$$\frac{\partial S_1}{\partial r} = \sqrt{2\Lambda + \frac{2\mu}{r} - \frac{\Gamma^2}{r^2} - \frac{2\mu^*}{r^3}} \rightarrow S_1(r) = \int_{r_0}^r \sqrt{2\Lambda + \frac{2\mu}{r} - \frac{\Gamma^2}{r^2} - \frac{2\mu^*}{r^3}} dr,$$

donde el límite inferior de integración r_0 será elegido como la mínima distancia entre el orbitador y el origen de coordenadas, o lo que es igual, la distancia en el periastro.

Hemos hallado, por tanto, unas expresiones para S_1 y S_2 , por lo que ahora podemos expresar la función generatriz S de la transformación como

$$S(r,\theta,\Lambda,\Gamma) = \Gamma\theta + \int_{r_0}^r \sqrt{2\Lambda + \frac{2\mu}{r} - \frac{\Gamma^2}{r^2} - \frac{2\mu^*}{r^3}} dr.$$
 (2.5)

De aquí en adelante, con objeto de simplificar las expresiones, llamaremos

$$\mathcal{R}(r) = \sqrt{2\Lambda + \frac{2\mu}{r} - \frac{\Gamma^2}{r^2} - \frac{2\mu^*}{r^3}},$$

y de ahí podremos poner

$$S(r,\theta,\Lambda,\Gamma) = \Gamma\theta + \int_{r_0}^{r} \mathcal{R}(r) dr.$$
 (2.6)

Significado de los nuevos momentos

Una vez obtenida la función generatriz, usaremos las expresiones dadas en (2.2) para obtener las relaciones que definen las nuevas variables canónicas.

Las dos primeras expresiones (2.2) nos conducen a las expresiones

$$R = \frac{\partial S}{\partial r} = \mathcal{R}(r) , \qquad (2.7)$$

$$\Theta = \frac{\partial S}{\partial \theta} = \Gamma. \tag{2.8}$$

De (2.8)se deduce que el segundo momento de las nuevas variables coincide, al igual que para las variables de Delaunay, con el segundo momento de las variables polares-nodales, esto es, con la norma del momento angular. De esta forma hemos dado significado a los nuevos momentos:

- Λ representa la energía del sistema, como se estableció en la página 20.

Obtención de las nuevas variables

Las dos últimas relaciones (2.2) se podrán poner como

$$\lambda = \frac{\partial S}{\partial \Lambda} = \int_{r_0}^{r} \frac{dr}{\mathcal{R}(r)},$$
 (2.9)

$$\gamma = \frac{\partial S}{\partial \Gamma} = \theta - \int_{r_0}^{r} \frac{\Gamma dr}{r^2 \mathcal{R}(r)}.$$
 (2.10)

En primer lugar, realizaremos el cambio de variables r = 1/z, que conduce a las expresiones

$$\lambda = -\int_{z_0}^{z} \frac{dz}{z^2 \sqrt{2\Lambda + 2\mu z - \Gamma^2 z^2 - 2\mu^* z^3}} , \qquad (2.11)$$

$$\gamma = \theta - \int_{z_0}^{z} \frac{\Gamma dz}{\sqrt{2\Lambda + 2\mu z - \Gamma^2 z^2 - 2\mu^* z^3}}$$
, (2.12)

 $donde z_0 = \frac{1}{r_0}.$

Las expresiones anteriores, una vez integradas y después de deshacer el cambio de variables, se podrán poner como:

$$\lambda = \lambda(r), \qquad \gamma = \theta - f(r).$$
 (2.13)

Obtención de las funciones $\lambda(r)$ y f(r).

La expresión de las funciones (2.11) y (2.12) se obtendrá mediante la integración de expresiones en las que aparece la raíz cuadrado de un polinomio de grado tres. Este tipo de expresiones conduce a integrales elípticas que pueden obtenerse haciendo uso de las relaciones que aparecen en el libro Byrde Friedman (1971).

Las expresiones de estas integrales se obtienen factorizando los polinomios y aplicando un determinado cambio de variables en función del valor relativo del extremo de integración con respecto a las raíces a, b y c del polinomio. Así, tendremos los siguientes casos:

En lo que sigue se hará uso de las siguientes integrales elípticas:

> Integral elíptica incompleta de primera especie

$$F(\boldsymbol{\varphi}, \boldsymbol{k}) = \int_0^{\varphi} \frac{d\theta}{\sqrt{1 - k^2 \operatorname{sen}^2 \theta}}.$$

> Integral elíptica completa de segunda especie

$$E(\boldsymbol{\varphi}, \boldsymbol{k}) = \int_0^{\varphi} \sqrt{1 - k^2 sen^2 \theta} \ d\theta.$$

> Integral elíptica incompleta de tercera especie

$$\prod(\boldsymbol{\varphi}, \boldsymbol{\alpha}^2, \boldsymbol{k}) = \int_0^{\varphi} \frac{d\theta}{(1 - \alpha^2 \operatorname{sen}^2 \theta) \sqrt{1 - k^2 \operatorname{sen}^2 \theta}}.$$

Así como de las funciones elípticas de Jacobisn(u,k), cn(u,k) y dn(u,k), definidas a partir de las siguientes relaciones

$$sn(u,k) = sen[am(u,k)].$$

 $cn(u,k) = cos[am(u,k)].$

$$dn(u,k) = \sqrt{1 - k^2 \operatorname{sen}^2[am(u,k)]}.$$

donde $\varphi = am(u, k)$ es la inversa de la integral elíptica de primera especie, es decir,

$$\psi = am(u, k) \quad \leftrightarrow \quad u = F(\psi, k)$$

Una detallada descripción de las propiedades de estas funciones puede verse también en *Byrd& Friedman* (1971).

A continuación se presenta la expresión de las distintas integrales, necesarias para el cálculo de (2.11) y (2.12)según el valor del límite de integración. En todos los casos se indica también el cambio de variables necesario para llegar a ese resultado y el número de la formula donde aparece en *Byrder Friedman (1971)*. Se escribe en negrita el resultado final y sin negrita algunos de los resultados intermedios de la aplicación de la transformación.

Caso 1.a(233.18)

$$\int_{c}^{y} \frac{dz}{z^{m} \sqrt{(a-z)(b-z)(z-c)}} = \frac{g}{c^{m}} \int_{0}^{u_{1}} \frac{du}{\left(1 - \frac{c-b}{c} sn^{2}u\right)^{m}}
= \frac{g}{c^{m}} \int_{0}^{u_{1}} \frac{du}{(1 - \alpha^{2} sn^{2}u)^{m}} \to m = 2 \to \frac{g}{c^{2}} \int_{0}^{u_{1}} \frac{du}{(1 - \alpha^{2} sn^{2}u)^{2}} = \frac{g}{c^{2}} V_{2}
= \frac{g}{c^{2}} \frac{1}{2(\alpha^{2} - 1)(k^{2} - \alpha^{2})} \left[\alpha^{2} E(u) + (k^{2} - \alpha^{2})u \right]
+ (2\alpha^{2}k^{2} + 2\alpha^{2} - \alpha^{4} - 3k^{2}) \prod(\varphi, \alpha^{2}, k) - \frac{\alpha^{4} sn(u) cn(u) dn(u)}{1 - \alpha^{2} sn^{2}u} \right],$$

siendo

$$\alpha = \frac{c - b}{c} \quad ; \quad k^2 = \frac{b - c}{a - c}; \quad g = \frac{2}{(a - c)^{\frac{1}{2}}} \quad ; \quad sn^2 u = \frac{z - c}{b - c}$$

$$\varphi = \sin^{-1} \left(\frac{y - c}{b - c}\right)^{\frac{1}{2}} \quad ; \quad u = F \left[\arcsin\left(\sqrt{\frac{z - c}{b - c}}\right), \sqrt{\frac{b - c}{a - c}}\right].$$

Caso 2.a(234.17)

$$\int_{y}^{b} \frac{dz}{z^{m} \sqrt{(a-z)(b-z)(z-c)}} = \frac{g}{b^{m}} \int_{0}^{u_{1}} \frac{dn^{2m}udu}{\left(1 - \frac{ak^{2}}{b}sn^{2}u\right)^{m}}$$

$$= \frac{g}{b^{m}} \int_{0}^{u_{1}} \frac{dn^{2m}udu}{(1 - \alpha^{2}sn^{2}u)^{m}} = \frac{\frac{g}{b^{m}}k^{2m}}{\alpha^{2m}} \sum_{j=0}^{m} \frac{(\alpha^{2} - k^{2})^{j}m!}{k^{2j}j!(m-j)!} V_{j} ,$$

$$m = 2 \qquad \rightarrow \qquad \frac{\frac{g}{b^{2}}k^{4}}{\alpha^{4}} \sum_{j=0}^{2} \frac{2(\alpha^{2} - k^{2})^{j}}{k^{2j}j!(2-j)!} V_{j}$$

con

$$\begin{split} V_0 &= \int du = u = F(\varphi,k) \; ; \; V_1 = \prod(\varphi,\alpha^2,k) \\ V_2 &= \frac{1}{2(\alpha^2-1)(k^2-\alpha^2)} \bigg[\alpha^2 E(u) + (k^2-\alpha^2) u \\ &+ (2\alpha^2 k^2 + 2\alpha^2 - \alpha^4 - 3k^2) \prod(\varphi,\alpha^2,k) - \frac{\alpha^4 sn(u) \; cn(u) \; dn(u)}{1-\alpha^2 sn^2 u} \bigg] \; , \end{split}$$

siendo

$$k^{2} = \frac{b-c}{a-c} \quad ; \quad g = \frac{2}{(a-c)^{\frac{1}{2}}} \quad ; \quad \alpha^{2} = \frac{ak^{2}}{b}; sn^{2}u = \frac{(a-c)(b-z)}{(b-c)(a-z)};$$

$$\varphi = \sin^{-1} \sqrt{\frac{(a-c)(b-y)}{(b-c)(a-y)}} u = F \left[\arcsin \left(\sqrt{\frac{(a-c)(b-z)}{(b-c)(a-z)}} \right), \sqrt{\frac{b-c}{a-c}} \right].$$

Caso 3.a(235.19)

$$\int_{b}^{y} \frac{dz}{z^{m} \sqrt{(a-z)(b-z)(z-c)}} = \frac{g}{b^{m}} \int_{0}^{u_{1}} \frac{dn^{2m}u \, du}{\left(1 - \frac{ck^{2}}{b} sn^{2}u\right)^{m}} = \frac{g}{b^{m}} \int_{0}^{u_{1}} \frac{dn^{2m}u \, du}{(1 - \alpha^{2} sn^{2}u)^{m}}$$

$$= \frac{g}{b^{m}} \frac{k^{2m}}{\alpha^{2m}} \sum_{j=0}^{m} \frac{(\alpha^{2} - k^{2})^{j} m!}{k^{2j} j! (m-j)!} V_{j}$$

$$m = 2 \qquad \Rightarrow \qquad \frac{gk^{4}}{b^{2} \alpha^{4}} \sum_{j=0}^{2} \frac{2(\alpha^{2} - k^{2})^{j}}{k^{2j} j! (2-j)!} V_{j} ,$$

con

$$\begin{split} V_0 &= \int du = u = F(\varphi,k) &; & V_1 = \prod(\varphi,\alpha^2,k) \\ V_2 &= \frac{1}{2(\alpha^2-1)(k^2-\alpha^2)} \bigg[\alpha^2 E(u) + (k^2-\alpha^2) u \\ &+ (2\alpha^2 k^2 + 2\alpha^2 - \alpha^4 - 3k^2) \prod(\varphi,\alpha^2,k) - \frac{\alpha^4 sn(u) \, cn(u) \, dn(u)}{1-\alpha^2 sn^2 u} \bigg] \;, \end{split}$$

siendo

$$sn^{2}u = \frac{(a-c)(z-b)}{(a-b)(z-c)} \; ; \quad k^{2} = \frac{a-b}{a-c} \; ; \quad g = \frac{2}{\sqrt{a-c}} \; ; \quad \varphi = \sin^{1}\sqrt{\frac{(a-c)(y-b)}{(a-b)(y-c)}}$$

$$\alpha^{2} = \frac{ck^{2}}{b} \quad ; \quad u = F\left[\arcsin\left(\sqrt{\frac{(a-c)(z-b)}{(a-b)(z-c)}}\right), \sqrt{\frac{a-b}{a-c}}\right].$$

Caso 4.a(236.19)

$$\int_{y}^{a} \frac{dz}{z^{m} \sqrt{(a-z)(b-z)(z-c)}} = \frac{g}{a^{m}} \int_{0}^{u_{1}} \frac{du}{\left[1 - \frac{a-b}{a}sn^{2}u\right]^{m}} = \frac{g}{a^{m}} \int_{0}^{u_{1}} \frac{du}{\left[1 - \alpha^{2}sn^{2}u\right]^{m}}$$

$$= \frac{g}{a^{m}} V_{m}$$

$$m = 2 \qquad \rightarrow \qquad \frac{g}{a^{2}} V_{2}$$

$$= \frac{g}{a^{2}} \frac{1}{2(\alpha^{2} - 1)(k^{2} - \alpha^{2})} \left[\alpha^{2}E(u) + (k^{2} - \alpha^{2})u\right]$$

$$+ (2\alpha^{2}k^{2} + 2\alpha^{2} - \alpha^{4} - 3k^{2}) \prod(\varphi, \alpha^{2}, k) - \frac{\alpha^{4}sn(u) cn(u) dn(u)}{1 - \alpha^{2}sn^{2}u},$$

siendo

$$sn^{2}u = \frac{a-z}{a-b} \quad ; \quad k^{2} = \frac{a-b}{a-c} \quad ; \quad g = \frac{2}{\sqrt{a-c}} \quad ; \quad \psi = \sin^{-1}\sqrt{\frac{a-y}{a-b}}$$

$$\alpha = \sqrt{a-\frac{b}{a}} \quad ; \quad u = F\left[\arcsin\left(\sqrt{\frac{(a-z)}{(a-b)}}\right), \sqrt{\frac{a-b}{a-c}}\right].$$

Caso 1.b(233.00)

$$\int_{c}^{y} \frac{d\mathbf{z}}{\sqrt{(\mathbf{a}-\mathbf{z})(\mathbf{b}-\mathbf{z})(\mathbf{z}-\mathbf{c})}} = g \int_{0}^{u_{1}} du = gu_{1} = g \cdot sn^{-1}(sin\psi, k) = \mathbf{g} \cdot \mathbf{F}(\boldsymbol{\psi}, \boldsymbol{k}),$$

siendo

$$g = \frac{2}{(a-c)^{\frac{1}{2}}}$$
 ; $\psi = \sin^{-1}\left(\frac{y-c}{b-c}\right)^{\frac{1}{2}}$; $k^2 = \frac{b-c}{a-c}$.

Caso 2.b(234.00)

$$\int_{y}^{b} \frac{dz}{\sqrt{(a-z)(b-z)(z-c)}} = g \int_{0}^{u_1} du = gu_1 = g \cdot sn^{-1}(sin\varphi,k) = g \cdot F(\varphi,k),$$

siendo

$$g = \frac{2}{(a-c)^{\frac{1}{2}}}$$
 ; $\varphi = \sin^{-1}\left(\frac{(a-c)(b-y)}{(b-c)(a-y)}\right)^{\frac{1}{2}}$; $k^2 = \frac{b-c}{a-c}$.

Caso 3.b(235.00)

$$\int_{b}^{y} \frac{dz}{\sqrt{(a-z)(b-z)(z-c)}} = g \int_{0}^{u_1} du = gu_1 = g \cdot sn^{-1}(sin\varphi,k) = g \cdot F(\varphi,k),$$

siendo

$$g = \frac{2}{(a-c)^{\frac{1}{2}}} \qquad ; \qquad \varphi = \sin^{-1}\left(\frac{(a-c)(y-b)}{(a-b)(y-c)}\right)^{\frac{1}{2}} \qquad ; \qquad k^2 = \frac{a-b}{a-c}.$$

Caso 4.b(236.00)

$$\int_{y}^{a} \frac{d\mathbf{z}}{\sqrt{(a-\mathbf{z})(b-\mathbf{z})(\mathbf{z}-c)}} = g \int_{0}^{u_{1}} du = gu_{1} = g \cdot sn^{-1}(sin\psi, k) = \mathbf{g} \cdot \mathbf{F}(\psi, \mathbf{k}),$$

siendo

$$g = \frac{2}{(a-c)^{\frac{1}{2}}}$$
 ; $\psi = \sin^{-1}\left(\frac{a-y}{a-b}\right)^{\frac{1}{2}}$; $k^2 = \frac{a-b}{a-c}$.

Significado de las nuevas variables

Aunque el problema queda resuelto una vez obtenidas las expresiones de las funciones $\lambda(r)$ y f(r), éstas no nos permiten comprender fácilmente el significado de estas variables y su relación con las variables clásicas de Delaunay.

Por un lado, si tenemos en cuenta que el primer momento de las variables de Delaunay coincide la velocidad radial, se tendrá la relación dr = R dt, que de acuerdo (2.7) se podrá poner como $dr = \mathcal{R}(r) dt$, y llevada a (2.9) conduce a

$$\lambda = \int_{r_0}^r \frac{dr}{\mathcal{R}(r)} = \int_T^t \frac{\mathcal{R}(r) dt}{\mathcal{R}(r)} = t - T, \qquad (2.14)$$

lo que nos indica que la primera variable es el tiempo relativo al periastro, ya que el instante T corresponde al instante de paso por el periastro.

Por otro lado, de acuerdo con la definición de momento angular se tiene que $r^2\dot{f}=\Theta=\Gamma$, o lo es igual $r^2df=\Gamma\,dt$, donde f representa la anomalía verdadera o ángulo entre la dirección del periastro y la del orbitador. Teniendo en cuenta esta relación y la relación anterior $dr=\mathcal{R}(r)\,dt$, y llevando ambas a (2.10) se tendrá

$$\gamma = \theta - \int_{r_0}^r \frac{\Gamma dr}{r^2 \mathcal{R}(r)} = \theta - \int_0^f df = \theta - f.$$
 (2.15)

que es una relación idéntica a la de las variables de Delaunay pero donde al anomalía verdadera tiene un valor diferente por el hecho de que la posición del periastro varía por la perturbación.

Así pues el significado de las nuevas variable es el siguiente

- \triangleright λ representa el tiempo (relativo al periastro)
- $\triangleright \gamma$ es similar del argumento del periastro.

Expresiones explícitas de la transformación

De acuerdo con las relaciones anteriores podemos encontrar fácilmente la relación explícita que nos da las nuevas variables canónicas $(\lambda, \gamma, \Lambda, \Gamma)$ en función de las antiguas (r, θ, R, Θ)

- $\lambda = \lambda(r)$,
- $\gamma = \theta f(r)$,
- $\bullet \quad \Lambda = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) \frac{\mu}{r} + \frac{\mu^*}{r^3},$
- $\Gamma = \Theta$,

donde las funciones $\lambda(r)$ y f(r) han sido calculadas anteriormente.

Suponiendo que la expresión $r = r(\lambda)$ representa la inversa de la función $\lambda = \lambda(r)$, para calcular las variables (r, θ, R, Θ) , conocidas las variables $(\lambda, \gamma, \Lambda, \Gamma)$, bastará aplicar las igualdades

$$\begin{array}{l} \succ r = r(\lambda), \\ \succ \theta = \gamma + f(r), \\ \succ R = \sqrt{2\Lambda + \frac{2\mu}{r} - \frac{P_2^2}{r^2} - \frac{2\mu^*}{r^3}}, \\ \succ \Theta = \Gamma. \end{array}$$

Para completar el proceso de cálculo de unas variables en función de otras es necesario el conocimiento de la expresión $r = r(\lambda) = r(t - T)$, o lo que es igual, la expresión de r en función del tiempo. Esta relación ya es complicada en el caso kepleriano, donde no puede ser obtenida si no es a partir de un método numérico o bien a través de desarrollos en serie de potencias de la excentricidad de la órbita que permiten invertir la ecuación de Kepler. Para el modelo dinámico tratado en esta memoria este proceso resulta mucho más complejo y será abordado en el siguiente capítulo donde se obtendrá la expresión equivalente, para este problema de la ecuación de Kepler, que llamaremos ecuación de Kepler generalizada.

 $^{^8}$ No se consideran la variable v y el momento N, puesto que ambos coinciden en los dos conjuntos de variables.

Integración del problema principal ecuatorial en las nuevas variables

Finalmente abordaremos el problema de la integración del modelo principalecuatorial del satélite para el que la función hamiltoniana, expresada en las nuevas variables será

$$\mathcal{H}(\lambda, \gamma, \nu, \Lambda, \Gamma, N) = \Lambda$$
,

por lo que las ecuaciones de Hamilton podrán ponerse como

$$\dot{\lambda} = \frac{\partial \mathcal{H}}{\partial \Lambda} = 1 \qquad ; \qquad \dot{\Lambda} = -\frac{\partial \mathcal{H}}{\partial \lambda} = 0,$$

$$\dot{r} = \frac{\partial \mathcal{H}}{\partial \Gamma} = 0 \qquad ; \qquad \dot{\Gamma} = -\frac{\partial \mathcal{H}}{\partial \gamma} = 0,$$

$$\dot{\nu} = \frac{\partial \mathcal{H}}{\partial N} = 0 \qquad ; \qquad \dot{N} = -\frac{\partial \mathcal{H}}{\partial \nu} = 0,$$

que integradas nos darán

$$\lambda = t - T$$
 ; $\gamma = \gamma_0$; $\nu = \nu_0$, $\Lambda = \Lambda_0$; $\Gamma = \Gamma_0$; $N = N_0$,

Donde $(\gamma_0, \nu_0, \Lambda_0, \Gamma_0, N_0)$ representa el valor constante de las variables en el instante inicial T.

3. Obtención de la distancia r en función del tiempo t. Ecuación de Kepler Generalizada.

Expresión de la relación entre r v t como un oscilador armónico perturbado

Como se ha dicho en el apartado anterior, para completar la definición de las variables del tipo "Delaunay" en el problema principal-ecuatorial, y la integración del movimiento orbital en estas variables, es necesario obtener la relación r =r(t-T). Para ello volveremos de nuevo a la expresión (1.14) del hamiltoniano del problema

$$H = \frac{1}{2} \left(R^2 + \frac{\Theta^2}{r^2} \right) - \frac{\mu}{r} + \varepsilon \frac{\sigma}{r^3}.$$

Si aplicamos a dicho hamiltoniano las ecuaciones de Hamilton correspondientes a las variables r, R se obtendrá

$$\frac{dr}{dt} = \frac{\partial H}{\partial R} = R,$$

$$\frac{dR}{dt} = -\frac{\partial H}{\partial r} = \frac{3\varepsilon\sigma}{r^4} + \frac{\Theta^2}{r^3} - \frac{\mu}{r^2},$$

de donde podemos poner

$$\frac{d^2r}{dt^2} = \frac{dR}{dt} = \frac{\Theta^2}{r^3} - \frac{\mu}{r^2} + \frac{3\varepsilon\sigma}{r^4} \,. \tag{3.1}$$

A continuación introduciremos el siguiente cambio de variables

$$u = \frac{1}{r} + \beta$$
 ; $r^2 \frac{df}{dt} = \Theta$; $v = \frac{du}{df}$; $\beta = -\frac{\mu}{\Theta^2}$

que permite obtener las siguientes relaciones

$$\frac{du}{dt} = -\frac{1}{r^2} \frac{dr}{dt} \to \frac{dr}{dt} = -r^2 \frac{du}{dt} = -r^2 \frac{du}{df} \frac{ds}{dt} = -\Theta \frac{du}{df} ,$$

$$\frac{d^2r}{dt^2} = -\frac{d\Theta}{dt} \frac{du}{df} - \Theta \frac{d}{dt} \left(\frac{du}{df}\right) = -\Theta \frac{d^2u}{df^2} \frac{ds}{dt} = -\Theta \left(\frac{\Theta}{r^2}\right) \frac{d^2u}{df^2} = -\Theta^2 (u - \beta)^2 \frac{d^2u}{df^2}$$

⁹ El resto de variables tienen derivada nula por lo que son todas constantes.

que llevadas a (3.1) nos dan para el término de la izquierda la expresión

$$\frac{d^2r}{dt^2} = -\Theta^2(u - \beta)^2 \frac{d^2u}{df^2},$$

mientras que para el término de la derecha de (3.1) se tendrá

$$\frac{\Theta^2}{r^3} - \frac{u}{r^2} + \varepsilon \frac{3\sigma}{r^4} = \Theta^2(u - \beta)^3 - u(u - \beta)^2 + \varepsilon [3\sigma(u - \beta)^4].$$

Igualando y dividiendo por $-\Theta^2(u-\beta)^2$ se tiene

$$\frac{d^2u}{df^2} = -(u - \beta) + \frac{u}{\Theta^2} + \varepsilon[3\sigma(u - \beta)^2],$$

Como $\beta = -u/\Theta^2$ podremos poner

$$\frac{d^2u}{df^2} + u = \varepsilon[3\sigma(u^2 - 2u\beta + \beta^2)],$$

y finalmente llegamos a la expresión

$$\frac{d^2u}{df^2} + u = \varepsilon(k_0 + k_1 u + k_2 u^2), \qquad (3.2)$$

donde $k_0 = 3\sigma\beta^2$; $k_1 = -6\sigma\beta$; $k_2 = 3\sigma$.

La expresión anterior corresponde a la ecuación de un oscilador armónico perturbado, del tipo dado en **(A.6)**, con una frecuencia w = 1, y donde el tiempo es la anomalía verdadera f. Para integrar dicha ecuación aplicaremos el método KBM del apéndice B.

Aplicación del método KBM

El método KBM consiste en ensayar una solución

$$u = \delta \cos \psi + \sum_{n>1} \frac{\varepsilon^n}{n!} u_n(\delta, \psi) = \sum_{n>0} \frac{\varepsilon^n}{n!} u_n(\delta, \psi) , (3.3)$$

donde cada uno de los $u_n(\delta, \psi)$ es una función 2π -periodica de ψ . De forma que tanto δ como ψ varían de acuerdo con la relación diferencial:

$$\frac{d\delta}{df} = \sum_{n\geq 1} \frac{\varepsilon^n}{n!} A_n(\delta) = \sum_{n\geq 0} \frac{\varepsilon^n}{n!} A_n(\delta), \quad A_0 = 0,$$
 (3.4)

$$\frac{d\psi}{df} = 1 + \sum_{n>1} \frac{\varepsilon^n}{n!} B_n(\delta) = \sum_{n>0} \frac{\varepsilon^n}{n!} B_n(\delta), \quad B_0 = 1.$$
 (3.5)

Observemos que la segunda relación nos indica que $\psi = f + \varepsilon \psi(\delta)$, esto es, representa una pequeña perturbación de la anomalía verdadera.

En primer lugar calcularemos los valores $F_{0,n}$ que conforman la ecuación que aparece indicada en el apéndice B (A.13):

$$F(u, v; \varepsilon) = \sum_{n>0} \frac{\varepsilon^n}{n!} F_{0,n} ,$$

 $\mathrm{donde}v = \frac{du}{df}$, término que no aparece en nuestra expresión.

Para hacerlo sustituimos en (3.2) los valores de u por el desarrollo en serie (3.3), donde si nos quedamos en el orden dos¹⁰ tendremos

$$u = u_0 + \varepsilon u_1 + \frac{\varepsilon^2}{2} u_2.$$

Por lo tanto

$$\frac{d^2u}{df^2} + u = F(u, v, \varepsilon)$$

$$= \varepsilon \left[k_0 + k_1 \left(u_0 + \varepsilon u_1 + \frac{\varepsilon^2}{2} u_2 \right) + k_2 \left(u_0 + \varepsilon u_1 + \frac{\varepsilon^2}{2} u_2 \right)^2 \right].$$

Si nos quedamos con los términos de ε^n tal que $n \leq 2$, tendremos finalmente

$$F(u, v, \varepsilon) = \varepsilon(k_0 + k_1 u_0 + k_2 u_0^2) + \varepsilon^2(u_1 k_1 + 2u_0 u_1 k_2).$$

Así que, emparejando términos obtenemos que

$$F_{01} = k_0 + k_1 u_0 + k_2 u_0^2$$
 ; $F_{02} = u_1 k_1 + 2u_0 u_1 k_2$. (3.6)

Tal y como viene explicado en el apéndice B, vamos a suponer que cada uno de los valores de u_n puede ser expresado mediante una expansión de serie de Fourier de una función 2π -periodica. Para ello comencemos calculando los términos U_n , mediante la aplicación de la fórmula (A.13)en la quenecesitaremos conocer tanto el valor de $F_{0,n}$ (valores hallados en (3.6))como el del w_n correspondiente.

¹⁰De aquí en adelante se obtendrá una teoría de orden 2. Para teorías de orden mayor la complejidad de las expresiones que aparecen se hace muy grande y es necesario usar procesadores simbólicos para su obtención.

Para hallar el valor de cada uno de los w_n haremos uso de la formula (A.11)que necesita de los valores de A_{n-1} , B_{n-1} y u_{n-1} , es decir, exactamente un orden inferior (por lo que son valores previamente hallados). Aplicando este proceso iterativo podemos calcular los valores de u_n hasta el orden que queramos.

Ahora vamos a establecer los valores iniciales. Tal y como aparece en **(A.8)** y **(A.9)**tenemos que $u_0 = \delta \cos \psi$, $A_0 = 0$ y $B_0 = 1$. De igual modo, se tiene que $w_0 = 0$, por lo que finalmente obtenemos que $U_0 = F_{0,0} - w_0 = 0$.

Continuamos del mismo modo, ahora para hallar los términos n = 1. Aplicando las mismas fórmulas que en el párrafo anterior, obtenemos que:

$$w_1 = 0$$

$$U_1 = F_{0,1} - w_1 = k_0 + k_1 u_0 + k_2 u_0^2 = k_0 + k_1 \delta \cos \psi + k_2 \delta^2 \cos^2 \psi,$$

Si como se dice en el apéndice B, llamamos $c_j(f)$ y $s_j(f)$ a los coeficientes respectivos de los términos en $cosj\psi$ y $senj\psi$ del desarrollo en serie de Fourier de la función $f(\psi)$ podremos poner finalmente, para U_1

$$c_0(U_1) = k_0 + \frac{k_2 \delta^2}{2}$$
 ; $c_1(U_1) = k_1 \delta$; $c_2(U_1) = \frac{k_2 \delta^2}{2}$.

Con los datos anteriores, y haciendo uso de las expresiones (A.15) y (A.16), tendremos que

$$A_1 = 0$$
,

$$B_1 = -\frac{k_1}{2},$$

$$c_0(u_1) = (k_0 + k_2 \delta^2/2),$$

$$c_1(u_1)=0,$$

$$c_2(u_1) = -\frac{k_2\delta^2}{6},$$

mientras que el resto de los coeficientes $c_n(u_1)$, $s_n(u_1)$ del desarrollo en serie de Fourier de u_1 son iguales a a cero. Finalmente la expresión de u_1 se podrá poner como

$$u_1 = \left(k_0 + \frac{k_2\delta^2}{2} - \frac{k_2\delta^2}{6}cos2\psi\right),\,$$

Siguiendo exactamente el mismo proceso calculamos ahora los términos de un orden superior. Comenzamos con el valor de $F_{0,2}$:

$$F_{0,2} = u_1 k_1 + 2u_0 u_1 k_2$$

Sustituyendo el valor de $u_{
m 1}$, previamente hallado, queda que

$$\begin{split} F_{0,2} &= \left(k_1 k_0 + \frac{k_1 k_2 \delta^2}{2}\right) + \delta k_2 (2 k_0 + k_2 \delta^2) cos \psi - k_1 k_2 \delta^2 \cos 2\psi \cos 2\psi \\ &- 2 k_2^2 \delta^3 \cos \psi \cos 2\psi \;. \end{split}$$

Tal y como se explica en el apéndice B, obtenemos el valor de w_2 por medio de la relación (A.11)

$$w_2 = 2(w_{110} + w_{101} + w_{011}),$$

donde, de acuerdo con(A.12)se tiene

$$w_{110} = -\frac{k_1^2 \delta cos \psi}{4}$$
 ; $w_{101} = w_{011} = -\frac{k_1 k_2 \delta^2}{3} cos 2\psi$,

por lo que al final podemos poner

$$w_2 = -\frac{k_1 \delta}{6} \left(3k_1 cos\psi + 8k_2 \delta cos2\psi \right).$$

Así que ya estamos en condiciones de hallar el valor de $U_2 = F_{0,2} - w_2$ que resulta ser

$$\begin{split} U_2 = \left(k_1 k_0 + \frac{k_1 k_2 \delta^2}{2}\right) + \delta k_2 \left(2k_0 + \frac{k_1^2}{2k_2} - \frac{5k_2 \delta^2}{6}\right) \cos\psi + 7k_1 k_2 \delta^2 \cos 2\psi \\ - k_2^2 \delta^3 \cos 3\psi \; . \end{split}$$

Continuando el proceso como en el orden anterior obtendremos

$$A_2=0,$$

$$B_2 = k_2 \left(\frac{5}{6} k_2 \delta^2 - 2k_0 - \frac{k_1^2}{2k_2} \right),$$

$$c_1(u_2)=0$$

$$c_2(u_2) = -\frac{7}{18}k_1k_2\delta^2 \ ,$$

$$c_3(u_2)=k_2^2\delta^3\;,$$

lo que permite expresar el valor de u_2 en la forma

$$u_2 = (k_1 k_0 + k_1 k_2 \delta^2) - \frac{7}{18} (k_1 k_2 \delta^2) \cos 2\psi + k_2 \delta^3 \cos 3\psi.$$

A partir de u_0 , u_1 y u_2 , obtenemos la expresión aproximada al orden dos de u, que será

$$\begin{split} u &= \delta cos\psi + \varepsilon \left(k_0 + \frac{k_2\delta^2}{2} - \frac{k_2\delta^2}{6}cos2\psi\right) \\ &+ \frac{\varepsilon^2}{2} \left((k_1k_0 + k_1k_2\delta^2) - \frac{7}{18}(k_1k_2\delta^2)cos2\psi + k_2\delta^3cos3\psi\right). \end{split}$$

De igual modo también nos interesa hallar la expresión de v. Para ello aplicamos la formula (A.10)

$$v_n = \sum_{i+j=n} \frac{n!}{i! \, j!} \left(A_i \frac{\partial u_j}{\partial \delta} + B_i \frac{\partial u_j}{\partial \psi} \right),$$

a partir de la condición inicial $v_0 = -\delta sen\psi$.

Para el orden uno se tendrá:

$$v_1 = \frac{1!}{1! \cdot 0!} \left(A_1 \frac{\partial u_0}{\partial \delta} + B_1 \frac{\partial u_0}{\partial \psi} \right) + \frac{1!}{0! \cdot 1!} \left(A_0 \frac{\partial u_1}{\partial \delta} + B_0 \frac{\partial u_1}{\partial \psi} \right),$$

por lo que

$$v_1 = k_1 \delta sen\psi + k_2 \delta^2 sen2\psi .$$

Para el orden dos tendremos:

$$\begin{split} v_2 &= \frac{2!}{2! \, 0!} \bigg(A_2 \frac{\partial u_0}{\partial \delta} + B_2 \frac{\partial u_0}{\partial \psi} \bigg) + \frac{2!}{1! \, 1!} \bigg(A_1 \frac{\partial u_1}{\partial \delta} + B_1 \frac{\partial u_1}{\partial \psi} \bigg) \\ &\quad + \frac{2!}{0! \, 2!} \bigg(A_0 \frac{\partial u_2}{\partial \delta} + B_0 \frac{\partial u_2}{\partial \psi} \bigg) \,, \end{split}$$

y finalmente

$$v_2 = k_2 \delta \left(2k_0 + \frac{k_1^2}{2k_2} - \frac{5}{6}k_2 \delta^2 \right) sen\psi + 4k_1 k_2 \delta^2 sen2\psi - k_2 \delta^3 sen3\psi \,.$$

Por lo tanto la expresión de v, hasta el orden dos, será

$$\begin{split} v &= -\delta sen\psi + \varepsilon (k_1\delta \ sen\psi + k_2\delta^2 \ sen2\psi) \\ &+ \frac{\varepsilon^2}{2} \bigg[k_2\delta \left(2k_0 + \frac{k_1^2}{2k_2} - \frac{5}{6} \, k_2\delta^2 \right) \, sen\psi + 4k_1k_2\delta^2 \, sen2\psi \\ &- k_2\delta^3 \, sen3\psi \bigg] \,. \end{split}$$

Finalmente si tenemos en cuenta la expresión (3.4) y los valores obtenidos para A_1 y A_2 , se tendrá que la derivada de δ respecto de f vale cero por lo que δ resulta ser una constante.

Por otro lado, puesto que las funciones B_1 y B_2 solo dependen de δ , y ésta es constante, las funciones B_1 y B_2 son también constantes, por lo que llevadas a (3.5) obtendremos

$$\frac{d\psi}{df} = \left(1 + \varepsilon B_1 + \frac{\varepsilon^2}{2} B_2\right) = n_{\psi}, \qquad (3.6)$$

siendo n_{ψ} una constante. Integrando ésta se tendrá finalmente

$$\psi = n_{\psi} f , \qquad (3.7)$$

de ahí, que la variable ψ que no es sino la anomalía verdadera ψ multiplicada por una constante próxima a la unidad será llamada *anomalía media generalizada*.

Relación de r con ψ y f

Si ahora tenemos en cuenta la relación entre r y u indicada en la página 32, $u = \frac{1}{r} + \beta$, podremos expresar r en función de ψ

$$\begin{split} \frac{1}{r} &= u - \beta = (\delta \cos \psi - \beta) + \varepsilon \left(k_0 + \frac{k_2 \delta^2}{2} - \frac{k_2 \delta^2}{6} \cos 2\psi \right) \\ &+ \frac{\varepsilon^2}{2} \left((k_1 k_0 + k_1 k_2 \delta^2) - \frac{7}{18} (k_1 k_2 \delta^2) \cos 2\psi + k_2 \delta^3 \cos 3\psi \right), \end{split}$$

por otro lado, si introducimos las siguientes igualdades

$$p=-rac{1}{eta}$$
 ; $e=rac{\delta}{eta}$; $a=rac{p}{1-e^2}$

dondep, e, a son respectivamente el semi-lado recto, la excentricidad y el semieje mayor, podremos poner

$$\begin{split} \frac{1}{r} &= \frac{1 + e cos\psi}{p} + \varepsilon \left(k_0 + \frac{k_2 \delta^2}{2} - \frac{k_2 \delta^2}{6} cos2\psi \right) \\ &+ \frac{\varepsilon^2}{2} \left(k_1 k_0 + k_1 k_2 \delta^2 - 7 k_1 k_2 \delta^2 cos2\psi + k_2 \delta^3 cos3\psi \right). \end{split}$$

Para obtener r haremos uso de las expresiones

$$\sum_{n\geq 0} \frac{\varepsilon^n}{n!} c_n = \frac{1}{\sum_{n\geq 0} \frac{\varepsilon^n}{n!} a_n},$$

donde los coeficientes de la serie pueden obtenerse de forma iterativa en la forma

$$c_0 = \frac{1}{a_0}, \quad c_n = -\frac{1}{a_0} \sum_{i=1}^n \binom{n}{i} a_i c_{n-i},$$

Aplicando dicha relación a 1/r, tendremos

$$c_0 = \frac{1}{a_0},$$

$$\begin{split} c_1 &= -\frac{p}{1 + e cos \psi} \sum_{i=1}^{1} {1 \choose i} a_i c_{i-1} \\ &= -\frac{p^2}{(1 + e cos \psi)^2} \left(k_0 + \frac{k_2 \delta^2}{2} - \frac{k_2 \delta^2}{6} cos 2\psi \right), \end{split}$$

$$\begin{split} c_2 &= -\frac{p}{1 + e cos \psi} \sum_{i=1}^2 \binom{n}{i} a_i c_{2-i} \\ &= -\frac{p^2}{(1 + e cos \psi)^2} \Bigg[-\frac{p}{(1 + e cos \psi)} \bigg(k_0 + \frac{k_2 \delta^2}{2} - \frac{k_2 \delta^2}{6} cos 2 \psi \bigg)^2 \\ &+ 2 \left(k_1 k_0 + \frac{k_1 k_2 \delta^2}{2} - 7 k_1 k_2 \delta^2 cos 2 \psi + k_2 \delta^2 cos 3 \psi \right) \Bigg] \,. \end{split}$$

Con lo que podremos expresar el valor de r en la forma

$$r = \frac{p}{1 + e \cos \psi} + \varepsilon \left[-\frac{p^2}{(1 + e \cos \psi)^2} \left(k_0 + \frac{k_2 \delta^2}{2} - \frac{k_2 \delta^2}{6} \cos 2\psi \right) \right]$$

$$+ \frac{\varepsilon^2}{2} \left[-\frac{p^2}{(1 + e \cos \psi)^2} \left[-\frac{p}{(1 + e \cos \psi)} \left(k_0 + \frac{k_2 \delta^2}{2} \right) \right. \right.$$

$$\left. -\frac{k_2 \delta^2}{6} \cos 2\psi \right)^2$$

$$+ 2 \left(k_1 k_0 + \frac{k_1 k_2 \delta^2}{2} - \frac{7}{18} k_1 k_2 \delta^2 \cos 2\psi + k_2 \delta^2 \cos 3\psi \right) \right].$$

Ecuación de Kepler generalizada

En el apartado anterior se ha obtenido la relación $r(\psi)$, por lo que para obtener la relación r(t), es necesario encontrar la relación $\psi(t)$.

Reuniendo las expresiones

$$\frac{d\psi}{df} = n_{\psi}$$
 ; $r^2 \frac{df}{dt} = \Theta$,

podemos encontrar la relación diferencial entre el tiempo y la anomalía verdadera generalizada ψ ,

$$n_{\psi}\Theta dt = r^2 d\psi , \qquad (3.8)$$

cuya integración permitirá obtener la relación que nos falta para determinar la función $\psi(t)$.

En primer lugar calculamos el valor de r^2 , que llevado a (3.8) transforma esta relación en la siguiente:

$$n_{\psi}\Theta dt = \frac{p^{2} \cdot d\psi}{(1 + e\cos\psi)^{2}} + \varepsilon \left[-\frac{2p^{3}}{(1 + e\cos\psi)} (k_{3} - k_{5}\cos2\psi) \right] d\psi$$

$$+ \frac{\varepsilon^{2}}{2} \left[\frac{4p}{1 + e\cos\psi} (k_{4} - k_{6}\cos2\psi + k_{7}\cos3\psi) + \frac{p^{4}}{(1 + e\cos\psi)^{4}} (k_{8} + k_{9}\cos2\psi + 2(k_{3} - k_{10}\cos2\psi)^{2}) \right] d\psi , \qquad (3.9)$$

donde

$$k_{3} = k_{0} + \frac{k_{2}\delta^{2}}{2},$$

$$k_{4} = k_{1}k_{0} + \frac{k_{1}k_{2}\delta^{2}}{2},$$

$$k_{5} = \frac{k_{2}\delta^{2}}{6},$$

$$k_{6} = \frac{7}{18}k_{1}k_{2}\delta^{2},$$

$$\begin{split} k_7 &= k_2 \delta^2 \;, \\ k_8 &= k_0^2 + \frac{2k_0 k_2 \delta^2}{2} + \frac{k_2^2 \delta^4}{4} \;, \\ k_9 &= \frac{k_2^2 \delta^4}{36} - \frac{2k_0 k_2 \delta^2}{6} - \frac{2k_2^2 \delta^4}{12} \;, \\ k_{10} &= \frac{k_2 \delta^2}{6} \;, \end{split}$$

En lo que sigue usaremos los polinomios T_n de Chebyshev de primer tipo¹¹, definidos a partir de la relación de recurrencia

$$T_0(x) = 1$$
, $T_1(x) = x$, $T_{n+1}(x) = 2 x T_n(x) - T_{n-1}(x)$.

Esta familia de polinomios tiene como propiedad fundamental:

$$\cos n\psi = T_n(\cos\psi),$$

lo que los hace especialmente útiles para su uso con funciones trigonométricas.

Si introducimos la variable q por medio de la relación

$$q = 1 + e \cos \psi$$
,

tendremos entonces que

$$\cos n\psi = T_n\left(\frac{q-1}{e}\right),\,$$

que llevada a **(3.9)** y después de realizar la integral en ambos lados, se obtiene, tras laboriosos cálculos, la expresión¹²

$$n_{\psi} \Theta (t - T) = A_{1} \int q^{-2} d\psi + \varepsilon \left(A_{3} \int q^{-1} d\psi - 2A_{3} \int q^{-2} d\psi + B_{1} \int q^{-3} d\psi \right)$$

$$+ \frac{\varepsilon^{2}}{2} \left(A_{9} \int q^{3} d\psi - 3A_{9} \int q^{2} d\psi + B_{2} \int q d\psi + B_{3} \int d\psi \right)$$

$$+ B_{4} \int q^{-1} d\psi + B_{5} \int q^{-2} d\psi + B_{6} \int q^{-3} d\psi$$

$$+ B_{7} \int q^{-4} d\psi \right) (5.7)$$

$$(3.10)$$

donde

¹¹ Los polinomios de Chebyshev de primer tipo son una familia de polinomios ortogonales definidos de forma recursiva.

 $^{^{12}}$ Todas las integrales de esta expresión son integrales definidas en el intervalo $[0,\psi]$ sin $\,$ embargo por simplicidad, de aquí en adelante, se han omitido los límites de integración en la expresión.

$$A_1 = p^2$$
 , $A_2 = -2p^3k_3$, $A_3 = \frac{2p^3k_2\delta^2}{e^2}$,

$$A_4=2p^4k_3^2+\frac{p^4k_2^2\delta^4}{18}+\frac{2}{3}p^4k_2k_3\delta^2+p^4k_3^2+\frac{k_2^2\delta^4p^4}{36}+\frac{k_3k_2\delta^2p^4}{3}\;,$$

$$A_5 = \frac{2}{9} \frac{p^4 k_2^2 \delta^4}{e^4} + \frac{1}{9} \frac{k_2^2 \delta^4 p^4}{e^4} \ , \qquad A_6 = -\frac{2}{9} \frac{p^4 k_2^2 \delta^4}{36 e^2} - \frac{4}{3} \frac{p^4 k_2 k_3 \delta^2}{e^2} - \frac{4 k_2^2 \delta^4 p^4}{36 e^2} \ ,$$

$$A_7 = 4pk_4 + \frac{14pk_1k_2\delta^2e^2}{9e^2} \ , \qquad \qquad A_8 = -\frac{29pk_1k_2\delta^2}{9e^2} \ ,$$

$$A_9=\frac{k_2\delta^2}{12e^3}\ , \qquad \qquad A_{10}=-\frac{3k_2\delta^2e^2}{48e^3}\ , \\ B_1=A_2+A_3\ , \qquad B_2=A_8+3A_9+A_{10}\ , \qquad B_3=A_5-2A_8-A_9-A_{10}\ ,$$

$$B_4 = -4A_5 + A_7 + A_8 , B_5 = 6A_5 + A_6 ,$$

$$B_6 = -4A_5 - 2A_6$$
 , $B_7 = A_4 + A_5 + A_6$.

La expresión (3.10) nos da la relación entre t y ψ , a partir de las integrales del tipo $\int q^n d\psi$, donde n puede ser positivo o negativo. Si tenemos en cuenta la expresión de q, cuando n > 0 el integrando resulta una expresión polinómica de funciones circulares por lo que la integral resulta sencilla de realizar, sin embargo cuando n < 0, el integrando se convierte en una expresión racional de funciones circulares, mucho más difícil de integrar. Esta dificultad, que aparece en muchos problemas de la Astrodinámica, se solventa con facilidad combinando el uso de la anomalía verdadera con la excéntrica a partir del cambio de variables correspondiente. En nuestro caso el cambio equivalente viene definido por

$$\cos \psi = \frac{\sqrt{1 - e^2} \sec E}{1 - e \cos E} \qquad ; \qquad \sin \psi = \frac{\cos E - e}{1 - e \cos E},$$

en la que ψ , E representan respectivamente las anomalías verdadera y excéntrica generalizadas. Usando estas relaciones encontramos que

$$\frac{p}{1+e\cos\psi}=a(1-e\cos E) \qquad ; \qquad d\psi=\frac{\sqrt{1-e^2}}{1-e\cos E} dE,$$

De esta forma las integrales de potencias de q se podrán poner como

$$\int q^n d\psi = \begin{cases} \int (1 + e \cos \psi)^n d\psi, & n \ge 0\\ \eta^{-2n-1} \int (1 - e \cos E)^{-n-1} dE, & n < 0 \end{cases}$$

donde
$$\eta = \sqrt{1 - e^2}$$
.

Así tendremos finalmente

$$\int q^{-4}d\psi = \eta^7 \int (1 - e \cos E)^3 dE,$$

$$\int q^{-3}d\psi = \eta^5 \int (1 - e \cos E)^2 dE,$$

$$\int q^{-2}d\psi = \eta^3 \int (1 - e \cos E) dE,$$

$$\int q^{-1}d\psi = \eta$$

$$\int d\psi = \psi,$$

$$\int qd\psi = \int (1 + e \cos \psi)d\psi,$$

$$\int q^2 d\psi = \int (1 + e \cos \psi)^2 d\psi,$$

$$\int q^3 d\psi = \int (1 + e \cos \psi)^3 d\psi.$$

Con lo que en todos los casos las integrales tienen la forma de un polinomio de funciones trigonométricas. Aunque estas integrales resultan sencillas de realizar, la cantidad de cálculos necesarios para completar la integración, y el volumen de los resultados nos impide escribir estos en el texto.

A pesar de todo nos fijaremos en la forma que adquiere dicha expresión observando, en primer lugar los términos de orden cero de (3.10). Escribiendo la parte no perturbada del orden cero tendremos que

$$n_{\psi} \Theta(\mathsf{t} - \mathsf{T}) = p^2 \int q^{-2} d\psi = p^2 \eta^3 \int (1 - e \cos E) dE = p^2 \eta^3 (E - e \sin E),$$

que puede ponerse también como

$$n(t-T) = (E - e \sin E),$$

lo que equivale a la ecuación de Kepler. Si consideramos todos los términos de la perturbación podemos deducir que (3.10) se transforma en

$$\ell = n(t - T) = E - e \operatorname{sen} E + \sum_{i \ge 1} \frac{\varepsilon^i}{i!} \left(\frac{e^j}{j!} \Upsilon_{ij}(E, \psi) \right)$$

que llamaremos ecuación de Kepler generalizada.

Conclusiones

En la presente memoria se estudia el problema principal-ecuatorial del satélite artificial, para ello se ha analizado la dinámica del movimiento orbital considerando la perturbación producida por el achatamiento de la Tierra considerando un modelo limitado a orbitas ecuatoriales o cuasi-ecuatoriales.

Mediante la aplicación de la ecuación de Hamilton-Jacobi se ha conseguido, partiendo del hamiltoniano en el que se considera los términos que expresan la no-esfericidad de la Tierra, obtener un nuevo conjunto de variables canónicas, similares en forma a las conocidas variables de Delaunay. Se ha formulado la relación explícita, tanto directa como inversa, de estas variables con las variables polares-nodales. Estas relaciones aparecen en función de integrales elípticas y funciones de Jacobi. Finalmente se ha procedido a integrar el problema principal-ecuatorial en estas nuevas variables.

Con objeto de invertir la expresión de las función $\lambda(r)$ se ha reformuladoel problema principal-ecuatorial como si se tratara de un oscilador armónico perturbado, y tras ser tratado con el método KBM se ha obtenido la expresión inversa de la anterior $r(\psi)$, donde ψ es la anomalía verdadera generalizada. Finalmente, se ha obtenido la ecuación de Kepler generalizada, similar a la ecuación de Kepler del problema no perturbado, que nos da la relación entre la variables ψ y λ , o lo que es igual, el tiempo t. Con ello se obtiene la relación der con t.

Apéndices

A continuación vamos a realizar una presentación teórica de las herramientas matemáticas que se han utilizado durante la realización del trabajo. Se presentan únicamente los resultados fundamentales sin demostración.

Apéndice A: Elementos de dinámica hamiltoniana y ecuación de Hamilton-Jacobi.

Supondremos un sistema dinámico formado por N puntos P_i , i=1,...,N, de masas m_i y cuya posición viene expresada, en un sistema inercial, por los vectores \mathbf{x}_i . La dinámica de este sistema de puntos viene descrita por el conjunto de ecuaciones resultante de la aplicación de la ecuación fundamental de Newton a cada una de las partículas:

$$F_i = \frac{d\mathbf{p_i}}{dt}, \qquad i = 1, ..., N, \quad siendo \, \mathbf{p_i} = m_i \dot{\mathbf{x_i}}.$$
 (A.1)

Los puntos P_i no se mueven libremente sino que pueden estar sujetos a una serie de relaciones funcionales entre los vectores de posición, llamadas *ligaduras*, del tipo:

$$f(x_1, x_2, ..., x_N; t) = 0.$$

En general un sistema de N puntos viene representado por 3N coordenadas. Cada ligadura permite expresar una cualquiera de estas coordenadas en función del resto, lo que reduce en una unidad el número de coordenadas independientes. Por ello, si el sistema tiene k ligaduras, entonces podrán introducirse n = 3N - k coordenadas independientes, $\mathbf{q} = (q_1, ..., q_n)$, de forma que podremos expresar las posiciones de las partículas como:

$$x_i = x_i(q, t), \qquad i = 1, ..., N.$$
 (A.2)

Este conjunto de coordenadas independientes se llama coordenadas generalizadas, y el espacio n-dimensional de coordenadas libres se llama espacio de configuración. Las derivadas de las coordenadas generalizadas $\dot{q} = (\dot{q}_1, ..., \dot{q}_n)$ son las velocidades generalizadas. El número n de coordenadas libres del sistema se llama número de grado de libertad.

Se llama energía cinética de un sistema dinámico a la función

$$T = \sum_{i=0}^{N} \frac{1}{2} m_i \dot{x}_i^2.$$

Para expresar la energía cinética en función de las coordenadas generalizadas tendremos en cuenta la relación (A. 2), cuya derivada será

$$\dot{\mathbf{x}}_i = \frac{\partial \mathbf{x}_i}{\partial \mathbf{q}} \frac{d\mathbf{q}}{dt} + \frac{\partial \mathbf{x}}{\partial t} = \phi_i(\mathbf{q}, \dot{\mathbf{q}}, t),$$

Por tanto vemos que la energía cinética de un sistema dinámico puede depender tanto de las coordenadas, como de las velocidades generalizadas, es decir, $T = T(q, \dot{q}, t)$.

Por otro lado, sea \mathbf{F} el conjunto de fuerzas que actúa sobre el sistema y supongamos que existe¹³ un escalar V, cuyo gradiente respecto de las coordenadas generalizadas coincide con $-\mathbf{F}$, esto es

$$\mathbf{F} = -\nabla V = \left(\frac{\partial V}{\partial q_1}, \dots, \frac{\partial V}{\partial q_n}\right).$$

Esta función V = V(q), que no depende de las velocidades generalizadas, será llamada energía potencial de un sistema dinámico.

Con todo lo anterior, se llama *función lagrangiana de un sistema dinámico* a la expresión

$$\mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) = T(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) + V(\boldsymbol{q}).$$

Puede demostrarse fácilmente que las ecuaciones del movimiento de un sistema dinámico se pueden expresar, en términos de la función lagrangiana, en la forma

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} = \frac{d}{dt} \left(\nabla_q \mathcal{L} \right) - \nabla_q \mathcal{L} = 0. \tag{A.3}$$

Estas ecuaciones son las llamadas ecuaciones de Lagrange.

Se definen los *momentos generalizados*, $p = (p_1, ..., p_n)$, a partir de las igualdades

$$\boldsymbol{p_i} = \frac{\partial \mathcal{L}}{\partial \dot{\boldsymbol{q}_i}}.$$

_

¹³ En este caso la fuerza se dice conservativa.

Estas funciones nos permiten expresar las velocidades generalizadas \dot{q}_t en la forma

$$\dot{\mathbf{q}}_{i} = \dot{\mathbf{q}}_{i}(\mathbf{q}, \mathbf{p}, t).$$

Al espacio 2n dimensional (q, p) se le llama **espacio fásico**.

Llamaremos *función hamiltoniana* o *hamiltoniano* \mathcal{H} a la transformación de Legrende de la función lagrangiana considerada como función de \dot{q} , esto es

$$\mathcal{H}(\boldsymbol{q},\boldsymbol{p},t) = \boldsymbol{p} \cdot \dot{\boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p},t) - \mathcal{L}(\boldsymbol{q},\dot{\boldsymbol{q}}(\boldsymbol{q},\boldsymbol{p},t),t).$$

Las ecuaciones de Newton (A.1) y el sistema de ecuaciones de Lagrange (A.3) son equivalentes a las ecuaciones:

$$\dot{\boldsymbol{q}} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{p}} = \nabla_{\boldsymbol{p}} \mathcal{H}$$
 ; $\dot{\boldsymbol{p}} = -\frac{\partial \mathcal{H}}{\partial \boldsymbol{q}} = -\nabla_{\boldsymbol{q}} \mathcal{H}$,

que son las llamadas *ecuaciones de Hamilton* del sistema. Así como las ecuaciones de Newton y de Lagrange representan sistemas de ecuaciones diferenciales de orden dos, las ecuaciones de Hamilton constituyen un sistema de orden uno.

Se puede utilizar una notación más compacta llamando $\mathbf{x} = (\mathbf{q}, \mathbf{p}) \in \mathbb{R}^n \times \mathbb{R}^n$ al vector de coordenadas y momentos por lo que el Hamiltoniano se expresará como

$$\mathcal{H}(\mathbf{x},t) = \mathcal{H}(\mathbf{q},\mathbf{p},t).$$

Con todo lo anterior, la evolución dinámica del sistema viene dada por las ecuaciones de Hamilton, que ahora se pueden expresar como

$$\dot{\mathbf{x}} = \mathcal{J}\nabla_{\mathbf{x}}\mathcal{H}$$
,

Donde $\mathcal J$ es la matriz antisimétrica

$$\mathcal{J} = \begin{pmatrix} 0_n & I_n \\ -I_n & 0_n \end{pmatrix} ,$$

Que verifica $\mathcal{J}^{-1} = \mathcal{J}^T = -\mathcal{J}$, y donde 0_n , I_n representan, respectivamente, las matrices nula y unidad de orden n.

La formulación hamiltoniana constituye una de las herramientas más potentes para el estudio de los sistemas dinámicos. Las propiedades geométricas del espacio fásico nos dan idea del comportamiento global del sistema, por lo que una buena elección de este espacio fásico nos simplifica su estudio. En lo que sigue se estudiarán las transformaciones canónicas que, por sus propiedades, se adaptan perfectamente a este propósito.

Sea la transformación del espacio fásico ψ : $\mathbb{R}^{2n} \to \mathbb{R}^{2n}$: $\mathbf{x} = (\mathbf{q}, \mathbf{p}) \to \mathbf{y} = (\mathbf{Q}, \mathbf{P})$, definida por las expresiones $\mathbf{y} = y(\mathbf{x}, t)$, que supondremos de clase $C^{(1)}$ y tal que $\det \Gamma \neq 0$ en el dominio (\mathbf{x}, t) que se considere, siendo Γ la matriz jacobiana

$$\Gamma = \mathbf{y}_{x} = \nabla_{x} \mathbf{y} = \begin{pmatrix} \frac{\partial y_{i}}{\partial x_{j}} \end{pmatrix} = \begin{pmatrix} \frac{\partial y_{1}}{\partial x_{1}} & \cdots & \frac{\partial y_{1}}{\partial x_{2n}} \\ \cdots & \cdots & \cdots \\ \frac{\partial y_{2n}}{\partial x_{1}} & \cdots & \frac{\partial y_{2n}}{\partial x_{2n}} \end{pmatrix}.$$

Una transformación ψ que satisface las condiciones anteriores se dice transformación canónica, si y solo si, existe una constante μ tal que se satisface la relación $\Gamma \mathcal{J} \Gamma^{T} = \mu \mathcal{J}$. La constante μ es llamada multiplicador de la transformación. Cuando $\mu = 1$, la transformación se llama transformación completamente canónica.

El conjunto de transformaciones canónicas tiene una estructura de grupo respecto a la composición de transformaciones. El elemento neutro de este grupo es la transformación identidad, mientras que la inversa de una transformación canónica de matriz jacobiana Γ y multiplicador μ es otra transformación canónica de matriz jacobiana Γ^{-1} y multiplicador $\frac{1}{\mu}$.

En lo que sigue, y por simplificar, consideraremos únicamente transformaciones canónicas y = y(x) que no dependan del tiempo t puesto que en este caso, que es el considerado en el ejemplo de la memoria, las propiedades se simplifican notablemente.

La propiedad más importante de las transformaciones canónicas nos dice que si aplicamos una transformación $\mathbf{y} = \mathbf{y}(\mathbf{x})$ a un sistema dinámico autónomo de hamiltoniano $\mathcal{H}(\mathbf{x})$ entonces las ecuaciones diferenciales del sistema mantienen la forma hamiltoniana, esto es, se pueden expresar como

$$\dot{\boldsymbol{y}} = \mathcal{J}\nabla_{\boldsymbol{y}}K(\boldsymbol{y}),$$

siendo $K(y) = \mu \mathcal{H}(x(y))$ el hamiltoniano transformado.

Aunque existen diversos métodos para obtener transformaciones canónicas, nos fijaremos aquí únicamente en las transformaciones de contacto. Para ello, supongamos que se busca una transformación de $(q,p) \rightarrow (Q,P)$.

Puede demostrarse que dada una función cualquiera S(q, P), llamada función generatriz, la transformación definida por las ecuaciones

$$p = \nabla_q S, \qquad Q = \nabla_P S,$$
 (A.4)

es una transformación completamente canónica.

Aunque S puede ser cualquier función, no todas nos definen una transformación canónica que sea útil para el estudio de nuestro sistema dinámico. Por ello haremos uso de la propiedad que nos dice que si S(q, P) satisface la ecuación de Hamilton-Jacobi

$$\mathcal{H}(\mathbf{q}, \nabla_{\mathbf{q}} S) = h, \tag{A.5}$$

dondeh es la energía del sistema, entonces las nuevas variables y momentos (Q, P) son constantes del sistema dinámico de hamiltoniano \mathcal{H} .

La búsqueda de una transformación canónica se convierte en la búsqueda de una función S que verifique la ecuación de Hamilton-Jacobi; o lo que es igual, la búsqueda de una integral primera de una ecuación en derivadas parciales.

Apéndice B: Método de Krylov-Bogoliubov-Mitropolsky.

Krylov y Bogoliubov (1947) desarrollaron un método analítico de resolución del problema de un oscilador armónico perturbado que fue ampliado y justificado por Bogoliubov y Mitropolsky (1961). Calvo (1971) aplica por primera vez dicho método al problema principal del satélite artificial, mientras que Caballero (1971) lo usa para tratar un modelo en el que se incluyen los armónicos zonales J_2 , J_3 y J_4 .

El método propuesto por Krylov, Bogoliubov y Mitropolskise usa para obtener aproximaciones asintóticas para cualquier ecuación no linear de segundo orden que tenga la siguiente forma:

$$\frac{d^2u}{dt^2} + w^2u = F(u, v; \varepsilon), \tag{A.6}$$

donde $v = \frac{du}{dt}$ y ε es un parámetro pequeño. Además, F debe poder ser expresada como una serie de potencias de ε cuyo orden cero es nulo, esto es

$$F(u,v;\varepsilon) = \varepsilon f(u,v;\varepsilon) = \sum_{n\geq 0} \frac{\varepsilon^n}{n!} F_{n,0}(u,v), \qquad F_{0,0}(u,v) = 0. \quad (A.7)$$

Cuando $\varepsilon = 0$, la ecuación diferencial corresponde a un oscilador armónico cuya solución se puede escribir como $u = \delta \cos \psi$ con una amplitud constante δ y un ángulo de rotación de fase uniforme $\psi = w t + \psi_0$.

El método KBM supone que la solución de la ecuación diferencial bajo estudio (A.7) puede ser expresada como un desarrollo asintótico de la siguiente forma:

$$u = \delta cos\psi + \sum_{n\geq 1} \frac{\varepsilon^n}{n!} u_n(\delta, \psi) = \sum_{n\geq 0} \frac{\varepsilon^n}{n!} u_n(\delta, \psi), \quad u_o = \delta cos\psi, \quad (A.8)$$

donde cada uno de los $u_n(\delta, \psi)$ es una función 2π -periodica de ψ . De igual manera, supondremos que tanto δ como ψ varían de acuerdo con la relación diferencial:

$$\frac{d\delta}{dt} = \sum_{n\geq 1} \frac{\varepsilon^n}{n!} A_n(\delta) = \sum_{n\geq 0} \frac{\varepsilon^n}{n!} A_n(\delta), \qquad A_0 = 0, \qquad (A.9)$$

$$\frac{d\psi}{dt} = w + \sum_{n\geq 1} \frac{\varepsilon^n}{n!} B_n(\delta) = \sum_{n\geq 0} \frac{\varepsilon^n}{n!} B_n(\delta), \quad B_0 = w.$$

Para la realización de los cálculos presentados en este apéndice haremos uso de la generalización de la *fórmula de Cauchy*

$$\prod_{n=1}^{k} \left(\sum_{i \geq 0} \frac{\varepsilon^{i}}{i!} a_{i}^{(n)} \right) = \sum_{i \geq 0} \frac{\varepsilon^{i}}{i!} \left(\sum_{j_{1} + j_{2} + \dots + j_{k}} \frac{i!}{j_{1}! j_{2}! \dots j_{k}!} a_{j_{1}}^{(1)} a_{j_{2}}^{(2)} \dots a_{j_{k}}^{(k)} \right).$$

Diferenciando (A.8) con respecto al tiempo, obtenemos

$$v = \frac{du}{dt} = \left(\sum_{n>0} \frac{\varepsilon^n}{n!} \frac{\partial u_n}{\partial \delta}\right) \frac{d\delta}{dt} + \left(\sum_{n>0} \frac{\varepsilon^n}{n!} \frac{\partial u_n}{\partial \psi}\right) \frac{d\psi}{dt},$$

dondesi sustituimos los valores de $\frac{d\delta}{dt}$ y $\frac{d\psi}{dt}$ dados en **(A.9)**, obtenemos

$$v = \sum_{n \ge 0} \frac{\varepsilon^n}{n!} v_n(\delta, \psi), \quad v_n(\delta, \psi) = \sum_{i+j=n} \frac{n!}{i! \, j!} \left(A_i \frac{\partial u_j}{\partial \delta} + B_i \frac{\partial u_j}{\partial \psi} \right) (\mathbf{A}. \, \mathbf{10})$$

$$v_0 = -w \, \delta \, \sin \psi \, .$$

Si diferenciamos de nuevo v con respecto del tiempo se tendrá

$$\frac{d^{2}u}{dt^{2}} + w^{2}u = \sum_{n\geq 0} \frac{\varepsilon^{n}}{n!} \left[w^{2}u_{n} + \sum_{i+j+k=n} \frac{n!}{i! j! k!} \left(A_{i} \frac{dA_{j}}{d\delta} \frac{\partial u_{k}}{\partial \delta} + A_{i} \frac{dB_{j}}{d\delta} \frac{\partial u_{k}}{\partial \psi} + A_{i} A_{j} \frac{\partial^{2}u_{k}}{\partial \delta^{2}} + B_{i} B_{j} \frac{\partial^{2}u_{k}}{\partial \psi^{2}} + 2A_{i} B_{j} \frac{\partial^{2}u_{k}}{\partial \delta \partial \psi} \right].$$

Considerando los valores de u_0 , A_0 , B_0 y particularizando en la expresión anterior todos los términos de i, j o k iguales an, encontramos los siguientes términos no nulos:

$$\begin{split} i &= n, \qquad j = 0, \qquad k = 0 \, \rightarrow \, 2A_i B_j \frac{\partial^2 u_k}{\partial \delta \partial \psi} = 2A_n B_0 \frac{\partial^2 u_0}{\partial \delta \partial \psi} = -2w A_n sin\psi, \\ i &= n, \qquad j = 0, \qquad k = 0 \, \rightarrow \, B_i B_j \frac{\partial^2 u_k}{\partial \psi^2} = B_n B_0 \frac{\partial^2 u_0}{\partial \psi^2} = -w B_n \delta cos\psi, \\ i &= 0, \qquad j = n, \qquad k = 0 \, \rightarrow \, B_i B_j \frac{\partial^2 u_k}{\partial \psi^2} = B_0 B_n \frac{\partial^2 u_0}{\partial \psi^2} = -w B_n \delta cos\psi, \\ i &= 0, \qquad j = 0, \qquad k = n \, \rightarrow \, B_i B_j \frac{\partial^2 u_k}{\partial \psi^2} = B_0 B_0 \frac{\partial^2 u_n}{\partial \psi^2} = -w^2 \frac{\partial^2 u_n}{\partial \psi^2}, \end{split}$$

Por lo tanto podremos escribir

$$\frac{d^2u}{dt^2} + w^2u = \sum_{n>0} \frac{\varepsilon^n}{n!} \left(w^2 \frac{\partial^2 u_n}{\partial \psi^2} + w^2 u_n - 2w A_n \sin \psi - 2w B_n \delta \cos \psi + w_n \right),$$

donde

$$w_n = \sum_{\substack{i+j+k=n\\i,i,k\neq n}} \frac{n!}{i!\,j!\,k!} w_{ijk} \,, \tag{A.11}$$

siendo

$$w_{ijk} = \left(A_i \frac{dA_j}{d\delta} \frac{\partial u_k}{\partial \delta} + A_i \frac{dB_j}{d\delta} \frac{\partial u_k}{\partial \psi} + A_i A_j \frac{\partial^2 u_k}{\partial \delta^2} + B_i B_j \frac{\partial^2 u_k}{\partial \delta^2} \right.$$

$$\left. + 2A_i B_j \frac{\partial^2 u_k}{\partial \delta \partial \psi} \right). \tag{A. 12}$$

Todo lo hallado previamente representa la parte izquierda de la ecuación (A.6) en función de $\delta y \psi$. La parte derecha de la misma ecuacion se puede obtener expandiendo como series de potencia de ε la función $F(u, v; \varepsilon)$ en la siguiente forma:

$$F(u,v;\varepsilon) = \sum_{n>0} \frac{\varepsilon^n}{n!} F_{n,0} \left(\sum_{n>0} \frac{\varepsilon^n}{n!} u_n , \sum_{n>0} \frac{\varepsilon^n}{n!} v_n , \right) = \sum_{n>0} \frac{\varepsilon^n}{n!} F_{0,n} , (A. 13)$$

 $\mathrm{donde} F_{0,n} = F_{0,n}(u_0,u_1,\ldots,u_{n-1},v_0,v_1,\ldots,v_{n-1})$

Igualando orden a orden ambas expresiones previas para cada $n \geq 1$ nos queda

$$w^2\left(\frac{\partial^2 u_n}{\partial \psi^2} + u_n\right) = 2wA_0 \sin\psi + 2wB_n\delta\cos\psi + U_n, \quad U_n = F_{0,n} - w_n, \quad (A. 14)$$

donde U_n solo depende de A_p , B_p , u_p , $p \le n - 1$.

Llamando $c_j(f)$ y $s_j(f)$ a los coeficientes respectivos de los términos en $cosj\psi$ y $sinj\psi$ del desarrollo en serie de Fourier de la función $f(\psi)$ podremos poner finalmente

$$u_n(\psi) = c_0(u_n) + \sum_{j \ge 1} c_j(u_n) \cos j\psi + s_j(u_n) \sin j\psi,$$

$$U_n(\psi) = c_0(U_n) + \sum_{j \ge 1} c_j(U_n) \cos j\psi + s_j(U_n) \sin j\psi.$$

Con esta notación, se sustituimos u_n , U_n de la expresión (A.14) por su expansión de Fourier e identificamos término a término, obtenemos

$$A_n = -\frac{s_1(U_n)}{2w}, \quad B_n = -\frac{c_1(U_n)}{2w\delta},$$
 (A. 15)

así como los coeficientes de la expansión de u_n de la forma

$$c_0(u_n) = \frac{c_0(U_n)}{w^2}, \qquad c_j(u_n) = \frac{c_j(U_n)}{w^2(1-j^2)}, \qquad s_j(u_n) = \frac{s_j(U_n)}{w^2(1-j^2)},$$

$$j \ge 2, \quad (A.16)$$

Los coeficientes $c_1(u_n)$, $s_1(u_n)$ quedan, por definición, indeterminados, y pueden ser elegidos iguales a cero.

Con todo lo anterior, al ser un proceso iterativo, podemos ir hallando orden a orden los diferentes coeficientes de la expansión de Fourier y podemos hallar el valor tanto de u como de v hasta el orden que deseemos.

6. Bibliografía

- ➤ **Abad, A. (1985)** Problema Estelar Triple Jerarquizado: Integración en unas nuevas variables. *Actas X Jornadas Hispano-lusas de Matemáticas*. *Pag: 115-121*.
- Abad, A. (2012). Astrodinámica. Editor Bubok Publishing S.L. España.
- ➤ Byrd, Paul F.Friedman, Morris D. (1971). Handbook of Elliptic Integrals for engineers and scientists. Second Edition. Springer-Verlag New York Heidelberg.
- ➤ Bogoliubov, N. N. and Mitropolsky, Y. A (1961). Asymptotic Method in the Theory of Nonlinear Oscillations. *Gordon and Breach, New York*.
- ➤ Caballero, J. A. (1975). Movimiento de un satélite artificial bajo la acción gravitatoria terrestre. Teoría de segundo orden en variables de Hill. *Ph.D. Thesis*, *University of Zaragoza*.
- ➤ Calvo, M. (1971). Aplicación del método de promedios al estudio del movimiento de satélites artificiales. *Ph.D. Thesis, University of Zaragoza*.
- ➤ Danby, J. M. A. (1988). Fundamentals of Celestial Mechanics. Willmann-Bell, Virginia.

- ➤ Deprit, A and Rom, A. (1970), "The main problem of artificial satellite theory for small and moderate eccentricities". Celest. Mech. And Dyn. Astr. 2, 166-206.
- Delaunay, C. (1860). Théorie du movement de la lune, Mem. 28; 29 (1867), Acad. Sci. France, Paris.
- ➤ **Jezewski, D.J (1983).** An analytic Solution for the J₂ Perturbed Equatorial Orbit. Celestial Mechanics. *Pag. 367-371*. *Reidel Publishing Company*.
- ➤ Kovalevsky, J. (1967). Introduction to Celestial Mechanics. Springer, New York.
- ▶ Martinusi, V & Gurfil, P (2013). Analytical solutions for J_2 -perturbed unbounded equatorial orbits. Celestial Mechanics & Dynamical Astronomy V ol 115, pag 35-57.
- Soop, E. M. (1994). Handbook of Geostationary Orbits. Kluwer Academic, London and Microcosm, Inc, California.
- ➤ Wertz, J.R. and Larson, W. J. (2010). Space Mission Analysis and Design. Microcosm Press, Hawthorne, CA and Springer, New York.

AUTOMATIC REDUCTION OF GEO SURVEY DATA



Lluís Canals Ros

INDEX

<u>INDEX6</u>
ABSTRACT6
INTRODUCTION6
GEO SURVEY DATA7
TYPES OF ORBITS7
SPACE SITUATIONAL AWARENESS7
SPACE SURVEILLANCE AND TRACKING
TFRM TOOLS SOFTWARE7
TRACKING OBJECTS7
AUTOMATIC REDUCTION OF GEO SURVEY DATA7
ALGORITHM PROCESS 780
CONCLUSIONS8
APPENDIX 1. NAVSTAR-201408131356.TXT9
APPENDIX 2. MAIN.PY9
APPENDIX 3. PUNTO.PY10
APPENDIX 4. GLOBAL_CAND.TXT10
REFERENCES

ABSTRACT

In order to collaborate to the discovery and orbit maintenance of space objects orbiting the Earth, new software for planning tracking observation and cross-correlation of survey observations was developed. The cross-correlation is used for establishing paths of true objects, removing outliers and possible false detection in the images taken. For this cross-correlation two methods are implanted in the software, the H- δ and the α - δ positions and both are compared with Apex software (Pulkovo Observatory) results. It has been seen that due to the construction of the algorithm the H- δ method is more efficient than the α - δ method and with similar results with Apex but much more faster.

INTRODUCTION

In 1930, Bernhard Schmidt, a German optician invented the Schmidt camera, also referred to as the Schmidt telescope. This camera had several derivative designs and in 1956 Joseph Nunn worked in collaboration with Dr. James G. Baker to design and manufacture a series of satellite tracking cameras called Baker-Nunn cameras (BNC). In order to maximize the satellites coverage and minimize the positional measurement error, 21 BNCs were placed all around the world. One of these BNCs was installed at "Real Instituto y Observatorio de la Armada" (ROA) in San Fernando, Cadiz in 1958.

In early 80s with the upcoming of new satellite tracking technologies the BNC program became obsolete andwas cancelled. The BNC was donated to ROA and conserved waiting for its unknown refurbishment that finally takes place many years later.

It was on September 23rd of 2009, the (partly) refurbished BNC saw first light at the ROA during atest of its mount and assessment of the corrector and lens quality. After very promising image-quality results done at ROA and after confirmed in "Observatori Astronòmic del Montsec" (OAdM) the Telescope Fabra ROA Montsec (TFRM) project was inaugurated n 16 Sep 2010. The new telescope designation stands for the two partner institutions of the project: Reial Acadèmia de Ciències i Arts de Barcelona (RACAB) - Observatori Fabra, and Real Instituto y Observatorio de la Armada (ROA), as well as the observing site: Observatori Astronòmic del Montsec (OAdM).



Fig. 1 TFRM at OAdM. Sliding roof half open and south gabling wall fully open

The TFRM, have the ability of being robotically and/or remotely commanded by a control software based on a client-server architecture via Instrument-Neutral Distributed Interface (INDI) device communication protocol.

These are other specifications after the refurbishment.

Moun	t	
Туре	Equatorial	
Motion	า	
RA and DEC	Digital servo drives with arbitrary drift rate	
Optics		
Design	Baker-Nunn design with field flattener	
	andmeniscus lenses	
Aperture	0.5m	
Focal ratio	f/0.96	
Scale	3.9 arcsec / pixel	
Mirror diameter	0.78m	
Field of view (FOV)	4.4º x 4.4º	
Spot size	20 μm	
Filter	Schott GG475	
Detector		
Sensor	Kodak KAF-16803	
Format	4Kx4K, 9μm	
QE	60% peak at 550nm	
Camera	FLI PL16803 cooled with glycol (ΔT=65°C)	

Table 1. Post-refurbished Fabra-ROA Baker-Nunn specifications.

Due to its FoV (4.4º x 4.4º) , its tracking at arbitrary α and δ rates and the CCD shutter commanding at will during the exposure, TFRM is extremely useful forthe participation in

Space Situational Awareness/Space Surveillance and Tracking (SSA/SST)observational programs.

TFRM is suitable to survey the entire visible geostationary belt from itslocation and can cover twice our entire visible geostationary belt in a 12 hoursnight. This survey is done with the telescope stopped, and can detect objects close to the Geosynchronous orbit (GEO) as well as in the Highly Elliptical Orbits (HEO). TFRM is also suitable to track objects in any orbit.

The object of the present work will be based on the development of software for planning the observations for tracking objects and capable of the automatic reduction of the GEO survey data.

GEO SURVEY DATA

TYPES OF ORBITS

The TFRM software can be applied to any orbit while talking about tracking objects or in GEO/HEO orbits while applying results of the survey data observations. These are the orbits around earth:

Low Earth Orbit (LEO):Is commonly accepted as between 200 and 2000km above the Earth's surface. Spacecraft in LEO make one complete revolution of the Earth in about 90 minutes (the "orbital period"). The International Space Station (ISS) orbits between is in a LEO about 400 km above the Earth's surface. The Space Shuttle and Hubble Space Telescope also operate in this region. The proximity to the Earth's surface provides a high level of detail and ability to detect distinct objects (known as "spatial resolution"). Most remote sensing satellites, as well as national security assets are placed in LEO.

Medium Earth Orbit (MEO): Is the region of space around the Earth above LEO and below geosynchronous orbit (35792km). The orbital periods range are from about two to 12 hours. The most common use for satellites in this region is for navigation, such as the United States' Global Positioning System (GPS) (20197km), Russian Glonass (19103km) and European Galileo (23223km) constellations. Communication satellites that cover the North and South Poles are also put in MEO.

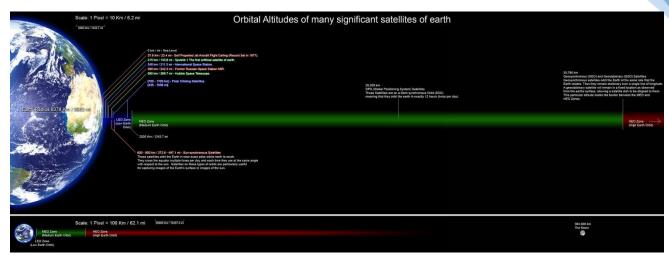


Fig.2 Schema of orbits around earth [2]

Geosynchronous/Geostationary Orbit (GEO):Is the region in which satellites orbit at approximately 35792kmabove the Earth's surface. At this altitude, the orbital period is equal to the period of rotation of the Earth, so the satellite appears to move neither East nor West. When a geosynchronous orbital path remains at the Earth's equator at all times, the satellite is in a geostationary orbit and therefore appears completely stationary relative to the surface of the Earth. This is convenient for communication satellites. In addition, geostationary satellites provide a "big picture" view, enabling coverage of weather events. However, instruments and sensors on satellites in geostationary orbit have poor spatial resolution due to the distance from the Earth.

Highly Elliptical Orbits (HEO): Are characterized by a relatively low-altitude (often under 1,000 km) perigee and an extremely high-altitude (often over 35,786 km) apogee. These extremely elongated orbits are useful for communication satellites because of long dwell times over a point in the sky; visibility near apogee can exceed 12 hours. For example, some satellite radio providers use HEO orbits to keep two satellites positioned above North America, while another satellite quickly sweeps through the southern part of its 24-hour orbit.

SPACE SITUATIONAL AWARENESS

With increasing dependence on space-based services, it is important to protect the space infrastructures, soSSA has become a prominent concern for both military and commercial systems, largely due to increasing military reliance on a range of space assets. Anti-satellite (ASAT) testing by China in 2007 and the 2009 collision of a non-operational Russian satellite with an operational Iridium satellite also raised concerns. Any damage to these space infrastructures could have significant consequences for the well-functioning of economic activities and our citizens' safety, among others. So, infrastructures are increasingly threatened by the risk of collision between spacecraft and more importantly, between spacecraft and space debris. Because of that, space debris has become the most serious threat to the sustainability of certain space activities.

The term Space Situational Awareness (SSA) refers to the ability to view, understand and predict the physical location of natural and manmade objects in orbit around the Earth, with the objective of avoiding collisions.

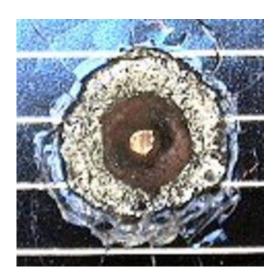


Fig. 3 Impact crater (size 4 mm) on solar cell retrieved from space [3]

SSA aims, to detect, predict and assess the risk to life and property due to man-made space debris objects, reentries, in-orbit explosions and release events, in-orbit collisions, disruption of missions and satellite-based service capabilities, potential impacts of Near-Earth Objects (NEOs), and the effects of space weather phenomena on space-based infrastructure.

SPACE SURVEILLANCE AND TRACKING

Space Surveillance and Tracking (SST) is the ability to detect and predict the movement of objects in orbit around the Earth. The data generated through an SST system can be used to actively protect space-based infrastructure, such as Earth observation satellites or navigation systems, from colliding with the ever-increasing cloud of man-made space debris.

Any SST system can be considered as a 'production line' for observation data. Sensors, such as telescopes, look at the sky and produce images of the Earth-orbiting objects that they see. These images are then transformed into plots that describe the path or trajectory of any particular object. Then, the plot must be examined to determine if it is showing a new object, or one already known to the system.

If the object is one that has already been seen, then the observations are used to update the record for that object. If the detected plot shows a new object, then the rest of the sensor network is used to try and see this newcomer again and obtain better data of its orbit. It is then added to the catalogue together with all the other observed objects.

In order to support the SST, there are two important catalogs. One is a Database and Information System Characterizing Objects in Space (DISCOS) is maintained by the European

Space Agency ESA since 1990[4]. DISCOS plays an essential role in the collision avoidance and re-entry prediction services provided by ESA's Space Debris Office, it is also providing input to numerous and very differently scoped engineering activities, within ESA and throughout industry. The other is the space-track catalogue of North American Aerospace Defense Command (NORAD)which will be explained in next chapter of this work.

TFRM is one of thesensors that contribute to the completeness of the objects without Two-Line-Element data of ESA'sDISCOS database, tracking known objects and collaborating in systematic surveys of the GEO Protected Zone since 2011.

TFRM TOOLS SOFTWARE

Software called "TFRM Tools" has been developed. It is implemented in Python 2.7.6[7], a high-level versatile object-orientedscripting programming language widely adopted by the scientific community. The Graphical User Interface (GUI) is implemented with PyQt4 [8], a cross-platform application and UI framework, which is fast and has great flexibility. Multithreading has been implemented in order for the application to remain responsive to input. The point is that in a single-threaded program, if the main execution thread blocks on a long-running task, the entire application can appear to freeze, so by moving such long-running tasks to a worker thread that runs concurrently with the main execution thread, it is possible for the application to remain responsive to user input while executing tasks in the background.Also CSS3, and HTML5 are supported for UI designing.

Some other libraries are used in the software in order to perform all the calculations and plots. The most relevant are listed and briefly explained below.

- Matplotlib: [9]
 - Matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.
- Peewee: [10]
 - Is a small, expressive Object-relational mapping (ORM)written in python built-in support for sqlite, mysql and postgresql and special extensions like <a href="https://example.com/http
- SciPy: [11]
 - SciPy is a collection of mathematical algorithms and convenience functions built on the Numpy extension of Python.
- PyEphem: [12]
 - PyEphem provides basic astronomical computations for the Python programming language.
- Other built-in python libraries used in the scripts:
 - Datetime, collections, cookielib, math, urllib, urllib2, sys, os, time.

In order to contribute to the Space Surveillance and Tracking, TFRM performs nightly surveys and tracking observations. TFRM Tools implements scripts for both types of observations that are explained in the next sections.

TRACKING OBJECTS

Preparing tracking observations could be done using this part of the software. In Fig. 4 is shown a view of the software, in particular the selected tab is the one used. Using this tab, we will be able to determine which satellites will be visible from the TFRM, that is to say, the next transit time of the satellite after the date and time introduced by the user, and elaborate an observation planning depending on the priorities. This planning is introduced in the INDI softwarewhich is commanding the TFRM to perform the scheduled observation.

To achieve this, the user has to enter some information in the program.

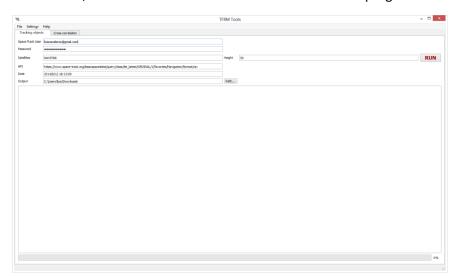


Fig. 4 Tracking objectstab view

In Fig. 5 is shown the fields that user has to fill in order to get the expected results. Fields are represented by numbers in order to refer to them.

Fields number 1 and number 2 are the username and password to auto login the space-track website [6]. This website promotes space flight safety, protection of the space environment and the peaceful use of space worldwide by sharing space situational awareness services and information with U.S. and international satellite owners/operators, academia and other entities. Using its Application Programming Interface (API) allows us to access data on this site programmatically using custom, stable URLs with configurable parameters. This API conforms to the general principles of a design called Representational State Transfer or "REST" and is identical to the data returned in the site's Graphical User Interface (GUI).

	ব্য	Т	FRM Tools
	File Settings	Help	
	Tracking objects	Cross-correlation	
ł	Space-Track User	lluiscanalsros@gmail.com]
1	Password	••••••]
ŧ	3 Satellites	NAVSTAR	Height 4 50
Ė	API	https://www.space-track.org/basicspacedata/query/class/tle_latest/ORDINAL/1/favorites/Navigation/format/csv]
ϵ	5 Date	2014/8/12 18:13:09]
ţ	7 Output	C: \Users\\liuis\Downloads	Path

Fig. 5 Fields to be filled by user

In field number 3, user can introduce the name of the satellites or leave it in blank if information of all of them wants to be retrieved.

Field number 4 is a restriction for those satellites selected. It is in degrees and in the example of the fig. 5 represents that the user wants all the satellites that its transit is above 50 degrees.

In field number 5 is where the API of the space-tack website is introduced. In this case user wants to retrieve the latest two line elements (TLE) for the navigation satellites in csv format. Any data retrieved have to be in csv extension file in order to be processed later by the script.

The TLE is a format specified by North American Aerospace Defense Command (NORAD), aformat for transmitting satellite Keplerian elements. The following is an example of a TLE with an explanation of the format:

```
ISS (ZARYA)
1 25544U 98067A 08264.51782528 -.00002182 00000-0 -11606-4 0 2927
2 25544 51.6416 247.4627 0006703 130.5360 325.0288 15.72125391563537
```

Line 1

Field	Columns	Content	Example
1	01-01	Line number	1
2	03-07	Satellite number	25544
3	08-08	Classification (U=Unclassified)	U
4	10-11	International Designator (Last two digits of launch year)	98
5	12-14	International Designator (Launch number of the year)	067
6	15–17	International Designator (Piece of the launch)	A
7	19–20	Epoch Year (Last two digits of year)	08
8	21–32	Epoch (Day of the year and fractional portion of the day)	264.51782528
9	34-43	First Time Derivative of the Mean Motion divided by two [2]	00002182
10	45-52	Second Time Derivative of Mean Motion divided by six (decimal point assumed)	00000-0
11	54-61	BSTAR drag term (decimal point assumed) [2]	-11606-4
12	63-63	The number 0 (Originally this should have been "Ephemeris type")	0
13	65–68	Element set number. incremented when a new TLE is generated for this object. [2]	292
14	69-69	Checksum (Modulo 10)	7

Line 2

Field	Columns	Content	Example
1	01–01	Line number	2
2	03–07	Satellite number	25544
3	09–16	Inclination [Degrees]	51.6416
4	18–25	Right Ascension of the Ascending Node [Degrees]	247.4627
5	27–33	Eccentricity (decimal point assumed)	0006703
6	35–42	Argument of Perigee [Degrees]	130.5360
7	44–51	Mean Anomaly [Degrees]	325.0288
8	53–63	Mean Motion [Revs per day]	15.72125391
9	64–68	Revolution number at epoch [Revs]	56353
10	69–69	Checksum (Modulo 10)	7

Field number 6 is for the date and time. It is the time from which all the calculations start, so the next transit after this time of each satellite is calculated and if it exists and is over the height restriction of field number 4 this satellite is represented in the final results.

The final results are written in a ".txt" file and in field number 7 users can choice where to save this generated file.

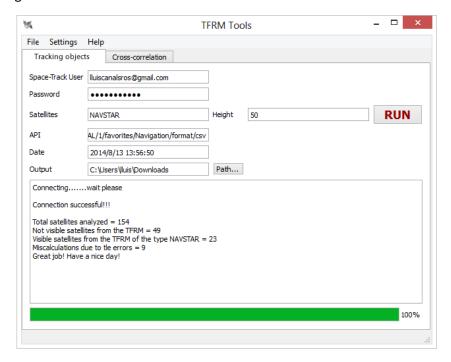


Fig. 6 Final results shown in the text browser embedded in the GUI

In fig. 6 is shown a summary of the final results. In this case the program has analyzed 154 satellites and 49 of them are below 50 degrees in its next transit. In this case user wants those satellites called NAVSTAR, so 23 of them will be visible above 50 degrees in the next transit.

The program saves a file in the output path with the default name "NAVSTAR-201408131356.txt". In this file is stored the name of the satellite, the Two Line Elements (TLE), the time when it is at its maximum height as well as its azimuth. In appendix 1 is a summary of this file.

With this obtained data, the user can prepare and elaborate a schedule for the observation in the INDI software, which is going to conduct the observation.

AUTOMATIC REDUCTION OF GEO SURVEY DATA

After the survey has been done, the next step is to process all the obtained images in order to get the space objects.

Fig. 7 is an example of the obtained image where space objects are inside green circles, easily identifiable among the trailed background stars. Exposure time is chosen to be long enough to allow one to distinguish between space objects and field stars and to achieve reasonable sensitivity, but not verylong to avoid producing extremely long trails of both field stars and space objects, which impacts accuracy and reliability of data reduction.



Fig.7 A 10s exposure taken with the TFRM

Next step is to get the position of those objects and discriminate them automatically from the background stars, and do it for all the images taken all over the night. This is done by Dr. Jorge Nuñez scripts based on the some ".dll" of the PinPoint software and SExtractor algorithm. In the future it will be implemented by TFRM Tools software. This process store these reduced points in a ".txt" extension file (Appendix 4) with the date, time, x and y positions on the CCD camera, right ascension (α), declination (δ), hour angle (H) and magnitude of each detection.

After all space objects are reduced, possible tracks have to be correlated and eliminate false detections. TFRM Tools use two methods for the cross-correlation, and user can choose between them.

- 1. Cross-correlation using hour angle (H) and declination (δ)
- 2. Cross-correlation using right ascension (α) and declination (δ)

For this purpose the Cross-correlation tab (Fig.8) the TFRM Tools software implements that.

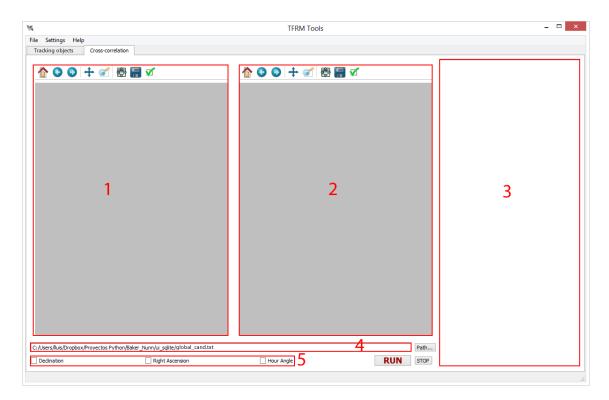


Fig. 8 Cross-correlation tab view

This tab is composed of:

- Number 1 and number 2 areembedded widgets from the matplotlib library to plot the total points retrieved by the image processing phase (left) and the final results after running the cross-correlation algorithm (right).
- Number 3 is a text browser to show the final tracks calculated by the algorithm.
- Number 4 is the path of the file user is going to use. This file contains the total
 points retrieved by the image processing phase in a file "global_cand.txt"
 extension file. This file is going to be converted into a ".csv" extension
 temporary file by the program in order to insert the data to the SQLite
 database.

 Number 5 is three checkboxes that user can choose between them. Only the combinations Declination-Right Ascension and Declination-Hour Angle are allowed.

The algorithm process can be separated in two parts. One is for the Declination-Right Ascension option, and the other is for the Declination-Hour Angle option. Both options have most of the code in common. In the next section is explained how the algorithm works.

ALGORITHM PROCESS

The algorithm can be divided in 5 steps:

- 1. Data treatment to be loaded into the database.
- 2. Construction of vectors of pairs of points linked by velocity.
- 3. Construction of a dictionary (key-value) of all possible tracks and iterate through this dictionary to find a track.
- 4. Remove points of positive tracks of the database and start again step number 2.
- 5. Plot the results in the matplotlib widget in the GUI and print the tracks found.

1 - Data treatment to be loaded into the database

In order to speed up the whole process calculations, the technology used by databases has been adopted. In particular the one used in TFRM Tools is SQLite. SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. Because of this, data retrieved from the .txt file (number 4 Fig. 4) has to be treated and converted to a ".csv" extension file to be loaded into the database with the correct structure (Fig. 9). The fields are, the id associated with the point, the image to which the point correspond, the date and time of the image, the coordinates (x, y) in the CCD, and the hour angle (ha), right ascension (ar), declination (dec) and magnitude (mag).

	id	image	hora	x	у	ha	ra	dec	mag
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	1	2011-05-25 00:32:00	1705.651	476.403	10.04055	14.544292	-6.400498	10.106
2	2	1	2011-05-25 00:32:00	305.138	1505.622	10.113924	14.470918	-4.883638	11.689
3	3	1	2011-05-25 00:32:00	1597.405	1955.878	10.147307	14.437535	-6.270643	9.745

Fig. 9 Summary of the database structure

In order to interact with the SQLite database an Object-relational mapping (ORM) library written in python and named Peewee is used. To make all the readings and writings from and to the database this is located in memory and destroyed when the thread ends.

2 - Construction of vectors of pairs of points linked by speed

Once we have all the data loaded in the database, data is prepared to be processed. The first filter applied in this algorithm is the velocity comparison, i.e., every point is linked with all other points of the following images that pass this velocity filter. Obviously one point cannot be related with other of the same image. In Fig. 10 is shown a summary output of this filter.

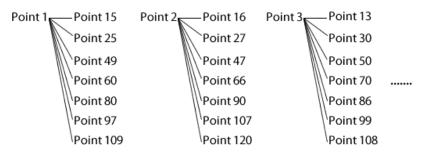


Fig. 10 Graphical schema output of the velocity comparison filter

All the iterations along all the registers of the database are done by SQLite engine, so these calculations are optimized by the database queries.

3 - Construction of a dictionary (key-value) of all possible tracks and iterate through this dictionary to find a track

Next step is, using these pairs of points linked by velocity, to construct a dictionary (key-value), because it allows us to rapidly iterate through every possible combination of paths. When the dictionary is created, all possible paths are evaluated and have to meet two constraints:

- Tracks with less than 5 detections are discarded.
- Linear correlation coefficient of the path must be high.

To construct the possible tracks a technique called recursion has been used in the algorithm. Fig. 10 is an example of a four images series and represents all possible tracks linked by velocity, i.e., first graphic represents that point number 1 is linked with points number 4, 5 and 6, and 4 is linked with points 7 and 8, and so on. First track to be analyzed is [1, 4, 7, 10], so the Pearson product-moment correlation coefficient is calculated for this track. If it fails next track to be analyzed will be [1, 4, 7, 11], and will continue until one tracks verifies the condition for the Pearson product-moment correlation coefficient imposed by user, and all the points of this track are deleted because will not be possible to form another track with any of these points.

Applying the recursion through the created dictionary allow the script to be generalized to any number of input images, because using the iteration method, conditions us, to a known number of images and make impossible to use the same script with another set of images. During the recursion over the dictionary, if the linear correlation coefficient of a track is inside the accepted limits imposed by the user, all those points are removed from the

dictionary in order not to be evaluated again, because one point cannot belong to more than one path.

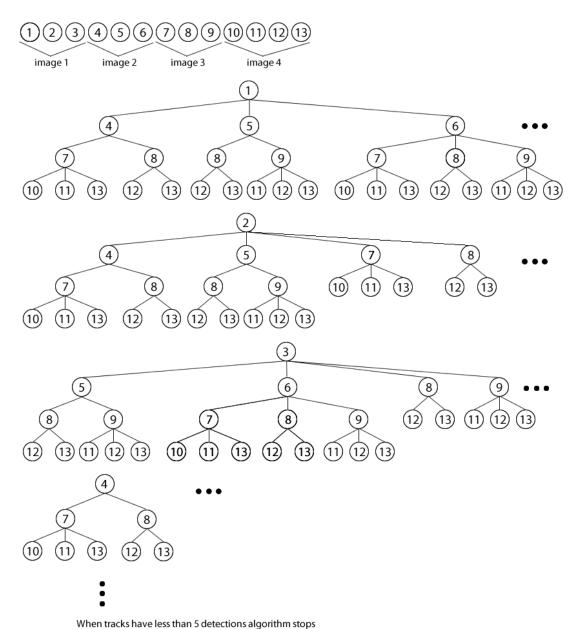


Fig.10 Schema of the recursion technique to find possible tracks

For the cross-correlation, The Pearson product-moment correlation is a measure of the strength of a linear association between two variables. Basically, a Pearson product-moment correlation attempts to draw a line of best fit through the data of two variables, and the Pearson correlation coefficient, indicates how far away all these data points are to this line of best fit (how well the data points fit this new model/line of best fit).

$$r = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}}$$

The Pearson correlation coefficient, can take a range of values from +1 to -1. A value of 0 indicates that there is no association between the two variables. A value greater than 0 indicates a positive association; that is, as the value of one variable increases, so does the value of the other variable. A value less than 0 indicates a negative association; that is, as the value of one variable increases, the value of the other variable decreases. The stronger the association of the two variables, the closer the Pearson correlation coefficient, will be to either +1 or -1 depending on whether the relationship is positive or negative, respectively. Achieving a value of +1 or -1 means that all your data points are included on the line of best fit - there are no data points that show any variation away from this line. Values between +1 and -1 (for example, 0.8 or -0.4) indicate that there is variation around the line of best fit. The closer the value to 0 the greater the variation around the line of best fit.

4 - Remove points of positive tracks of the database and start again step number 2

When recursion over the dictionary ends, and some tracks have been evaluated as positive, those tracks are removed from the initial table of the database and inserted to another table of this database. Initial table is called "Points" and the final table is called "Finaltracks".

Once those tracks are removed the process starts again with a higher velocity filterand step number 2 is started again. This iteration is done 3 times and helps to speed up the whole process since every iteration may have fewer points to be evaluated.

5 - Plot the results in the matplotlib widget in the GUI and print the tracks found

When the process finishes, all the points of the matched tracks are saved in the "Finaltracks" table of the database and those tracks are plotted on the right matplotlib widget of the GUI. In fig. 11 and 12are shown the final results of the algorithm for both filters implemented in the software, fig. 11 for the hour angle and declination filter and fig. 12 for the right ascension and declination filter. Each final track found is identified with a number and printed in the text browser widget in the right part of the GUI. It is also printed all the points that corresponds to each track as well as its hour angle, declination and right ascension.

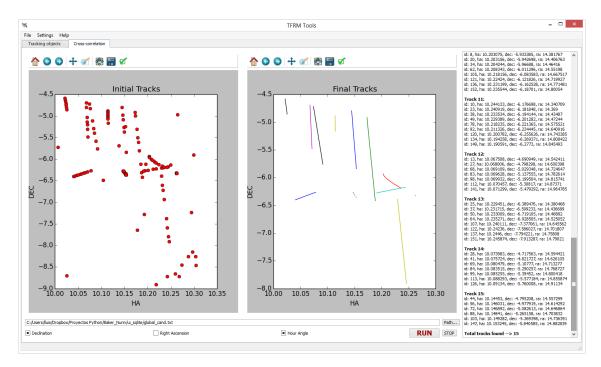


Fig. 11 Final results filtered by hour angle and declination

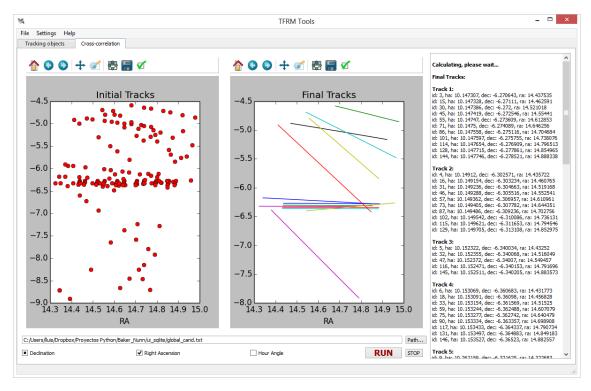


Fig. 12 Final results filtered by right ascension and declination

CONCLUSIONS

TFRM Tools software has begun to be developed in order to be an alternative to Apex utilities used in the TFRM.

In table 2 is shown a comparison of tracks found between the declination(δ)-hour angle(H) method, the right ascension(α)-declination(δ) method and the Apex software results. With the H- δ method the total tracks found are 15, with the α - δ method are 13 and with the APEX software are 16.

Track number 10, and number 14 are not found with the α - δ method. Furthermore, track number 3 and number 11 with the α - δ method some points of the track found by the H- δ method are not found.

The H- δ method works better than the α - δ method at least for this set of data. It adjusts better with the Apex software results. Id points with the "?" symbol in table 2 means that these pointsare not in the global_cand.txt file (Appendix 4) because the reduction of the images are done by Dr. Jorge Nuñez scripts based on the some ".dll" of the PinPoint software and SExtractor algorithm, a different method than the one done by Apex, so cannot appear as points in the final tracks. Another discrepancy with Apex results is Track number 10 wheretwo points are missing in TFRM Tools H- δ method and it is due to the curvature of this track.

As can be seen in table 2, all tracks detected by Apex are also found by TFRM Tools except Track 16 rejected because lack of enough points in clobal_cand.txt file.

An important point to remark is the execution time. While Apex is spending 30 minutes to make the correlation, TFRM Tools takes only 3 seconds in both methods.

In summary the cross-correlation step in TFRM Tools is comparable in performance to Apex in the tracks found but much more faster.

	Н-δ	α-δ	Apex α-δ
Track 1	id: 3	id: 3	id: 3
	id: 15	id: 15	id: 15
	id: 30	id: 30	id: 30
	id: 45	id: 45	id: 45
	id: 55	id: 55	id: 55
	id: 71	id: 71	id: 71
	id: 86	id: 86	id: 86
	id: 101	id: 101	id: 101
	id: 114	id: 114	id: 114
	id: 128	id: 128	id: 128

	id: 144	id: 144	id: 144
Track 2	id: 4	id: 4	id: 4
	id: 16	id: 16	id: 16
	id: 31	id: 31	id: 31
	id: 46	id: 46	id: 46
	id: 57	id: 57	id: 57
	id: 73	id: 73	id: 73
	id: 87	id: 87	id: 87
	id: 102	id: 102	id: 102
	id: 115	id: 115	id: 115
	id: 129	id: 129	id: 129
	-	-	?
Track 3	id: 5	id: 5	id: 5
	id: 17	id: 17	id: 17
	id: 32	id: 32	id: 32
	id: 47	id: 47	id: 47
	id: 58	id: 58	id: 58
	id: 74	-	id: 74
	id: 89	-	id: 89
	id: 104	id: 104	id: 104
	id: 116	id: 116	id: 116
	id: 130	id: 130	id: 130
	id: 145	id: 145	id: 145
Track 4	id: 6	id: 6	id: 6
	id: 18	id: 18	id: 18
	id: 33	id: 33	id: 33
	-	-	?
	id: 59	id: 59	id: 59
	id: 75	id: 75	id: 75
	id: 90	id: 90	id: 90
	-	-	?

	id: 117	id: 117	id: 117
	id: 131	id: 131	id: 131
	id: 146	id: 146	id: 146
Track 5	id: 9	id: 9	id: 9
	id: 24	id: 24	id: 24
	id: 35	id: 35	id: 35
	id: 51	id: 51	id: 51
	id: 63	id: 63	id: 63
	id: 80	id: 80	id: 80
	id: 95	id: 95	id: 95
	id: 106	id: 106	id: 106
	-	-	?
	id: 140	id: 140	id: 140
	id: 154	id: 154	id: 154
Track 6	id: 2	id: 2	id: 2
	id: 14	id: 14	id: 14
	id: 29	id: 29	id: 29
	id: 43	id: 43	id: 43
	id: 54	id: 54	id: 54
	id: 70	id: 70	id: 70
	id: 85	id: 85	id: 85
	id: 100	id: 100	id: 100
	-	-	?
	-	-	?
	id: 143	id: 143	id: 143
Track 7	id: 39	id: 39	id: 39
	id: 52	id: 52	id: 52
	id: 65	id: 65	id: 65
	id: 81	id: 81	id: 81
	id: 96	id: 96	id: 96
	id: 110	id: 110	id: 110

	id: 125	id: 125	id: 125
	-	-	?
Track 8	id: 1	id: 1	id: 1
	id: 12	id: 12	id: 12
	id: 26	id: 26	id: 26
	id: 40	id: 40	id: 40
	id: 53	id: 53	id: 53
	id: 67	id: 67	id: 67
	id: 82	id: 82	id: 82
	id: 97	id: 97	id: 97
	id: 111	id: 111	id: 111
	-	-	?
	id: 142	id: 142	id: 142
Track 9	id: 7	id: 7	id: 7
	id: 19	id: 19	id: 19
	-	-	?
	id: 48	id: 48	id: 48
	id: 61	id: 61	id: 61
	id: 76	id: 76	id: 76
	id: 91	id: 91	id: 91
	-	-	?
	id: 119	id: 119	id: 119
	id: 132	id: 132	id: 132
	id: 148	id: 148	id: 148
Track 10	id: 8	-	id: 8
	id: 20	-	id: 20
	id: 34	-	id: 34
	id: 62	-	id: 62
	-	-	id: 77
	-	-	id: 93
	id: 105	-	id: 105

	id: 121	-	id: 121
	id: 136	-	id: 136
	id: 152	-	id: 152
Track 11	id: 10	id: 10	id: 10
	id: 23	id: 23	id: 23
	id: 38	id: 38	id: 38
	id: 49	-	id: 49
	-	-	?
	id: 78	-	id: 78
	id: 92	-	id: 92
	-	-	?
	id: 120	id: 120	id: 120
	id: 134	id: 134	id: 134
	id: 149	id: 149	id: 149
Track 12	-	-	?
	id: 13	id: 13	id: 13
	id: 27	id: 27	id: 27
	-	-	?
	-	-	?
	id: 68	id: 68	id: 68
	id: 83	id: 83	id: 83
	id: 98	id: 98	id: 98
	id: 112	id: 112	id: 112
	-	-	?
	id: 141	id: 141	id: 141
Track 13	-	-	?
	id: 25	id: 25	id: 25
	id: 37	id: 37	id: 37
	id: 50	id: 50	id: 50
	id: 64	id: 64	id: 64
	-	-	?

	id: 107	id: 107	id: 107
	id: 122	id: 122	id: 122
	id: 137	id: 137	id: 137
	id: 151	id: 151	id: 151
Track 14	id: 28	-	id: 28
	id: 41	-	id: 41
	-	-	?
	id: 69	-	id: 69
	id: 84	-	id: 84
	id: 99	-	id: 99
	id: 113	-	id: 113
	id: 126	-	id: 126
	-	-	?
Track 15	-	-	?
	id: 44	id: 44	id: 44
	id: 56	id: 56	id: 56
	id: 72	id: 72	id: 72
	id: 88	id: 88	id: 88
	id: 103	id: 103	id: 103
	-	-	?
	-	-	?
	id: 147	id: 147	id: 147
Track 16	-	-	?
	-	-	108
	-	-	124
	-	-	139
	-	-	153
Total tracks found	15	13	16

Table 2. Comparison of tracks found between both methods.

APPENDIX 1. NAVSTAR-201408131356.txt

This is a summary of the complete output file.

```
0 OPS 5111 (NAVSTAR 1)
1 10684U 78020A 14225.03377787 .00000035 00000-0 00000+0 0 7616
2 10684 062.7643 184.2599 0076606 190.8724 015.8269 01.98072203250733
Date/Time (UTC)
             Máxima Altura Azimut
2014/8/13 21:32:19......63.9832450502......18.0750489119
_____
0 OPS 5114 (NAVSTAR 4)
1 11141U 78112A 14223.08457167 .00000039 00000-0 00000+0 0 9752
2 11141 062.5487 197.4441 0096830 024.3238 153.7751 01.92895015145740
Date/Time (UTC)
              Máxima Altura Azimut
2014/8/14 03:06:41.....71.9555374721......267.281940563
_____
0 OPS 9794 (NAVSTAR 8)
1 14189U 83072A 14224.10521360 .00000037 00000-0 00000+0 0 7722
2 14189 062.1506 194.1899 0138384 032.3415 136.8241 01.91718979143848
              Máxima Altura Azimut
Date/Time (UTC)
2014/8/14 03:17:50.....65.7211047222.....272.297485075
_____
0 NAVSTAR 11 (USA 10)
1 16129U 85093A 14223.90896427 .00000045 00000-0 00000+0 0 1564
2 16129 062.7093 204.2392 0198723 042.6376 087.1361 01.89368606147463
              Máxima Altura Azimut
Date/Time (UTC)
2014/8/13 23:55:47......68.4764030334......40.8005106675
_____
0 NAVSTAR 13 (USA 35)
1 19802U 89013A 14224.35748902 .00000029 00000-0 00000+0 0 1804
2 19802 054.4944 241.1515 0040078 140.6137 287.9538 01.85321879178970
Date/Time (UTC) Máxima Altura Azimut
2014/8/13 23:55:23......74.4059019878......346.860392621
_____
0 NAVSTAR 20 (USA 63)
1 20724U 90068A 14224.44022584 .00000031 00000-0 00000+0 0 2750
2 20724 053.7467 226.6614 0046052 264.1767 206.7052 01.87182503168930
Date/Time (UTC) Máxima Altura Azimut
2014/8/14 00:52:55......77.2694451369......26.6629736045
_____
0 NAVSTAR 21 (USA 64)
1 20830U 90088A 14224.27783383 .00000005 00000-0 00000+0 0 2494
2 20830 053.9758 160.4756 0018125 203.2519 284.9883 01.87754514171621
Date/Time (UTC)
              Máxima Altura Azimut
2014/8/13 20:05:08......75.6969716654......19.1476703569
0 NAVSTAR 22 (USA 66)
1 20959U 90103A 14224.21352315 .00000028 00000-0 00000+0 0 4593
2 20959 054.2946 206.7676 0114341 352.3768 007.4240 02.00563227173655
Date/Time (UTC)
              Máxima Altura Azimut
2014/8/13 18:48:22......88.6426185202......121.983053142
_____
```

APPENDIX 2. main.py

```
# -*- coding: utf-8 -*-
__license__ ='GPL v.3'
_author_ ="Lluis Canals Ros <lluiscanalsros@gmail.com>"
__version__ ='0.4'
from PyQt4 import QtCore, QtGui
from matplotlib.backends.backend qt4agg import FigureCanvasQTAgg as FigureCanvas
from matplotlib.backends.backend qt4agg import NavigationToolbar2QTAgg as
NavigationToolbar
from matplotlib.figure import Figure
import peewee as pw
import datetime as dt
from scipy import stats
from Punto import Punto
import ephem
import cookielib
import math
import urllib
import urllib2
from playhouse.csv_loader import*
import sys
import os
import collections as c
import time
db= pw.SqliteDatabase(None)
classPoints(pw.Model):
"""Importamos los datos a evaluar"""
id= pw.IntegerField(primary_key=True)
image= pw.IntegerField()
hora= pw.DateTimeField()
   x =pw.FloatField()
    y =pw.FloatField()
ha= pw.FloatField()
ra= pw.FloatField()
dec= pw.FloatField()
mag= pw.FloatField()
classMeta:
database= db
classFinaltracks(pw.Model):
"""Base de datos de resultados finales"""
id= pw.IntegerField(primary_key=True)
image= pw.IntegerField()
hora= pw.DateTimeField()
   x =pw.FloatField()
    y =pw.FloatField()
ha= pw.FloatField()
ra= pw.FloatField()
dec= pw.FloatField()
mag= pw.FloatField()
classMeta:
database= db
classTrackFinder(object):
total_tracks =[]
    @classmethod
defcalc_max_ha(cls, dift, vel_ha):
 """Calculo lo maximo que puede variar el angulo horario"""
        max_ha = (dift * vel_ha)/(3600*15)
return max ha
    @classmethod
defcalc_max_dec(cls, dift, vel_dec):
"""Calculo lo maximo que puede variar el declinacion"""
max dec = (dift * vel dec)/3600
```

```
return max_dec
@classmethod
defcalc_max_ra(cls, dift, vel_ra):
"""Calculo lo maximo que puede variar el declinacion"""
       max ra = (dift * vel ra)/3600
return max ra
defvelocity_comparing(self, candidates, vel_ha, vel_dec, pair_points):
"""Formo vector con pares de puntos conectados por velocidad""
for candidate in candidates:
for comp in Points.select().where(Points.hora > candidate.hora):
#calculo diferencia tiempo entre registros
dift= dt.timedelta.total seconds(comp.hora - candidate.hora)
if abs(candidate.ha - comp.ha) <= self.calc max ha(dift, vel ha) and \</pre>
abs(candidate.dec - comp.dec) <= self.calc_max_dec(dift, vel_dec):</pre>
# print self.calc_max_dec(dift, vel_dec)
                    p1 =Punto(candidate.id, candidate.ha, candidate.dec, candidate.ra)
                    p2 =Punto(comp.id, comp.ha, comp.dec, comp.ra)
                    pair points.append([p1, p2])
return pair_points
defvelocity comparing ra dec(self, candidates, vel ra, vel dec, pair points):
 ""Formo vector con pares de puntos conectados por velocidad""
for candidate in candidates:
for comp in Points.select().where(Points.hora > candidate.hora):
# calculo diferencia tiempo entre registros
dift= dt.timedelta.total seconds(comp.hora - candidate.hora)
if abs(candidate.ra - comp.ra) <= self.calc_max_ra(dift, vel_ra) and \</pre>
abs(candidate.dec - comp.dec) <= self.calc_max_dec(dift, vel_dec):
                    p1 =Punto(candidate.id, candidate.ha, candidate.dec, candidate.ra)
                    p2 =Punto(comp.id, comp.ha, comp.dec, comp.ra)
                    pair points.append([p1, p2])
return pair_points
    @classmethod
defcreate_dict(cls, dict_of_points, pair_points):
"""Creo el diccionario de putnos a partir del vector de pares de puntos conectados por
velocidad"""
for k, v in pair points:
            dict_of_points[k].append(v)
return dict_of_points
    @classmethod
defremove_points(cls, dis, leaf):
"""Si el track pasa la prueba de correlacion lineal elimino todos los puntos del
diccionario"""
for p in leaf:
if p in dis:
dis.pop(p)
return dis
deftest track(self, dis, track to correlate, final tracks, corrcoef, lap):
  'Testeo si el posible track pasa la prueba de correlacion lineal"
ha_try =[]
        dec try =[]
for pt in track to correlate:
            ha try.append(pt.ha)
            dec try.append(pt.dec)
        b =stats.pearsonr(ha_try, dec_try)
if-1< b[0]<-corrcoef[lap]or corrcoef[lap]< b[0]<1:</pre>
            final tracks.append(track_to_correlate)
            self.total tracks.append(track to correlate)
            self.remove_points(dis, track_to_correlate)
returnTrue
else:
returnFalse
defcollect tracks (self, dis, node, track to correlate, final tracks, corrcoef, lap):
        track to correlate.append(node)
# Cojo el vector de los puntos asociados a cada nodo (id de un punto)
print str(dis.get(node,[]))
if len(dis.get(node,[]))>0:
for n in dis[node]:
                is track = self.collect tracks(dis, n, track to correlate, final tracks,
```

```
corrcoef, lap)
if is track:
break
if len(track_to_correlate)>0andnot is_track:
                track_to_correlate.pop()
if len(track_to_correlate)<5:</pre>
                track to correlate.pop()
returnFalse
else:
                is track = self.test track(dis, track to correlate, final tracks,
corrcoef, lap)
ifnot is track:
                    track to correlate.pop()
return is track
deftest_track_by_ra_dec(self, dis, track_to_correlate, final_tracks, corrcoef, lap):
"""Testeo si el posible track pasa la prueba de correlacion lineal"'
        dec try =[]
for pt in track to_correlate:
            ra_try.append(pt.ra)
            dec try.append(pt.dec)
        b =stats.pearsonr(ra try, dec try)
if-1< b[0]<-corrcoef[lap]or corrcoef[lap]< b[0]<1:</pre>
print b[0]
            final tracks.append(track to correlate)
            self.total tracks.append(track to correlate)
            self.remove_points(dis, track_to_correlate)
returnTrue
else:
returnFalse
defcollect_tracks_by_ra_dec(self, dis, node, track_to_correlate, final_tracks, corrcoef,
        track to correlate.append(node)
if len(dis.get(node,[]))>0:
for n in dis[node]:
                is_track = self.collect_tracks_by_ra_dec(dis, n, track_to_correlate,
final_tracks, corrcoef, lap)
if is_track:
break
if len(track_to_correlate)>0andnot is_track:
                track_to_correlate.pop()
if len(track_to_correlate)<5:</pre>
                track_to_correlate.pop()
returnFalse
else:
               is track = self.test track by ra dec(dis, track to correlate,
final_tracks, corrcoef, lap)
ifnot is_track:
                    track_to_correlate.pop()
return is track
    @classmethod
defremove_tracks_from_db(cls, db1, db2, final_tracks):
"""Elimino tracks encontrados de la tabla Points y los paso a otra finaltracks"""
for track in final tracks:
for point in track:
                point row =db1.select().where(db1.id == point.id)
                row_dicts =[{'id': field.id,'image': field.image,'hora': field.hora,'x':
field.x,'y': field.y,
'ha': field.ha,'ra': field.ra,'dec': field.dec,'mag': field.mag}
for field in point_row]
                db2.insert many (row dicts).execute()
db1.delete().where(db1.id == point.id).execute()
deffilter_points_by_ha_dec(self):
        x = -1
lap=-1
corrcoef=[0.70,0.99,0.99]
for j in range(3):
            final_tracks =[]
            images_time =Points.select(Points.hora).distinct()
            number images =Points.select(Points.hora).distinct().count()
            cut on = number images /2+1
```

```
time array =[]
             id to cut =[]
            ha to cut =[]
            dec_to_cut =[]
            ra_to_cut =[]
            pair points =[]
for kk in images time:
                 time_array.append(kk.hora)
            n = time array[cut on]
            cut id =Points.select(Points.id, Points.ha, Points.dec,
Points.ra).where(Points.hora == n)
for ka in cut id:
                id to cut.append(ka.id)
                 ha_to_cut.append(ka.ha)
                 dec to cut.append(ka.dec)
                 ra to cut.append(ka.ra)
            id_cut = id_to_cut[0]
            ha_cut = ha_to_cut[0]
            dec cut = dec to cut[0]
            ra cut = ra to cut[0]
            point to cut =Punto(id cut, ha cut, dec cut, ra cut)
x+=1
# Las velocidades máximas estan expresadas en grados/hora
            vel increase =[0.05,1,5]
            max_vel_ha = vel_increase[x]
max_vel_dec = vel_increase[x]
candidates = Points.select()
            self.velocity comparing (candidates, max vel ha, max vel dec, pair points)
#Cremaos el diccionario
             dict of points =[]
            dict of points =c.defaultdict(list)
            self.create_dict(dict_of_points, pair_points)
            list_of_keys =[]
for key in dict_of_points.iterkeys():
                 list_of_keys.append(key)
# for key in list of keys:
     print str(key) + ", ",
lap+=1
for key in list of keys:
# Detiene el bucle si la key(el punto) coincide con el punto en el que hay que cortar pq
va no
# puedo formar un track lo suficientemente largo
if key == point_to_cut:
break
                 track to correlate =[]
                 self.collect_tracks(dict_of_points, key, track_to_correlate,
final_tracks, corrcoef, lap)
# Elimino de la base de datos los puntos que forman parte de final_tracks
self.remove_tracks_from_db(Points, Finaltracks, final_tracks)
deffilter_points_by_ra_dec(self):
lap=-1
corrcoef=[0.99,0.9999,0.9999]
for j in range(3):
            final tracks =[]
            images_time =Points.select(Points.hora).distinct()
             time_array =[]
for image time in images time:
                 time array.append(image time.hora)
            number_images =Points.select(Points.hora).distinct().count()
            cut_on =(number_images /2)+1
            n = time array[cut on]
            cut id =Points.select(Points.id, Points.ha, Points.dec,
```

```
Points.ra).where(Points.hora == n)
            id_to_cut =[]
            ha_to_cut =[]
            dec_to_cut =[]
            ra to cut =[]
for ka in cut_id:
                id_to_cut.append(ka.id)
                ha_to_cut.append(ka.ha)
                dec to cut.append(ka.dec)
                ra_to_cut.append(ka.ra)
            id cut = id to cut[0]
            ha cut = ha to cut[0]
            dec cut = dec to cut[0]
            ra_cut = ra_to_cut[0]
            point to_cut =Punto(id_cut, ha_cut, dec_cut, ra_cut)
# Las velocidades máximas estan expresadas en grados/hora
            vel_increase = [0.05, 1, 5]
            max vel dec = vel increase[x]
            max vel ra = 2.5
candidates= Points.select()
            pair_points =[]
            self.velocity comparing ra dec(candidates, max vel ra, max vel dec,
pair points)
# Cremaos el diccionario
            dict_of_points =[]
            dict of points =c.defaultdict(list)
            self.create_dict(dict_of_points, pair_points)
            list of keys =[]
for key in dict_of_points.iterkeys():
                list_of_keys.append(key)
# for key in list of keys:
    print str(key) + ", ",
lap+=1
for key in list_of_keys:
if key == point_to_cut:
break
                track_to_correlate =[]
                self.collect tracks by ra dec(dict of points, key, track to correlate,
final tracks, corrcoef, lap)
# Elimino de la base de datos los puntos que forman parte de final_tracks
self.remove_tracks_from_db(Points, Finaltracks, final_tracks)
classMatplotlibWidget(QtGui.QWidget):
def__init__(self):
        QtGui.QWidget.__init__(self)
self.size =(4.0,4.0)
        self.dpi =100
        self.fig =Figure(self.size, dpi=self.dpi)
        self.canvas =FigureCanvas(self.fig)
self.canvas.setParent(self)
        self.toolbar =NavigationToolbar(self.canvas, self)
        self.vbox =QtGui.QVBoxLayout()
self.vbox.addWidget(self.toolbar)
self.vbox.addWidget(self.canvas)
self.setLayout(self.vbox)
defget figure(self):
return self.fig
defdraw(self):
self.canvas.draw()
classMainwindow(QtGui.QMainWindow):
```

```
mysignal = QtCore.pyqtSignal(int)
def init (self):
super(Mainwindow, self).__init__()
# Create the tabbed widget
        self.central =QtGui.QTabWidget(self)
self.setCentralWidget(self.central)
# Create a new tab
        self.tab_sat_tools =QtGui.QWidget()
        self.tab find tracks =QtGui.QWidget()
# Configure tab find tracks
        self.vbox =QtGui.QGridLayout(self.tab find tracks)
        self.horizontal layout =QtGui.QHBoxLayout()
        self.widget =MatplotlibWidget()
self.widget.setObjectName("widget")
        self.horizontal layout.addWidget(self.widget)
        self.widget 2 =MatplotlibWidget()
        self.widget 2.setObjectName("widget
        self.horizontal layout.addWidget(self.widget 2)
self.vbox.addLayout(self.horizontal_layout,0,0,1,1)
        self.text browser =QtGui.QTextBrowser(self.tab find tracks)
        self.text browser.setMaximumSize(QtCore.QSize(300,16777215))
        self.text browser.setObjectName("text browser")
self.vbox.addWidget(self.text browser,0,1,2,1)
        self.grid_layout_3 =QtGui.QGridLayout()
self.grid_layout_3.setObjectName("grid_layout_3")
        self.ra =QtGui.QCheckBox(self.tab find tracks)
self.ra.setMaximumSize(QtCore.QSize(16777215,17))
self.ra.setObjectName("ar")
        self.grid_layout_3.addWidget(self.ra,1,1,1,1)
        self.path button =QtGui.QToolButton(self.tab find tracks)
        self.path button.setObjectName("toolButton")
        self.grid_layout_3.addWidget(self.path_button,0,4,1,1)
        self.stop button =QtGui.QToolButton(self.tab find tracks)
        self.stop button.setObjectName("stop button")
        self.grid_layout_3.addWidget(self.stop button,1,4,1,1)
        self.dec =QtGui.QCheckBox(self.tab_find_tracks)
self.dec.setMaximumSize(QtCore.QSize(16777215,17))
self.dec.setObjectName("dec")
        self.grid layout 3.addWidget(self.dec,1,0,1,1)
        self.ha =QtGui.QCheckBox(self.tab_find_tracks)
self.ha.setMaximumSize(QtCore.QSize(16777215,17))
self.ha.setObjectName("ah")
        self.grid_layout_3.addWidget(self.ha,1,2,1,1)
self.push button =QtGui.QPushButton(self.tab find tracks)
        self.push button.setMaximumSize(QtCore.QSize(80,25))
        self.push button.setObjectName("push button")
        self.grid layout 3.addWidget(self.push button,1,3,1,1)
        self.line_edit =QtGui.QLineEdit(self.tab_find_tracks)
        self.line_edit.setMaximumSize(QtCore.QSize(16777215,17))
        self.line_edit.setObjectName("line_edit")
        self.grid layout 3.addWidget(self.line edit,0,0,1,4)
self.vbox.addLayout(self.grid layout 3,1,0,1,1)
        self.menubar =QtGui.QMenuBar(self)
self.menubar.setGeometry(QtCore.QRect(0,0,949,21))
self.menubar.setObjectName("menubar")
        self.statusbar =QtGui.QStatusBar(self)
self.statusbar.setObjectName("statusbar")
self.setStatusBar(self.statusbar)
# Configure tab sat tools
        self.gridLayout =QtGui.QGridLayout(self.tab sat tools)
self.gridLavout.setMargin(10)
self.gridLayout.setObjectName("gridLayout")
        self.label user =QtGui.QLabel(self.tab sat tools)
        self.label_user.setObjectName("label_user")
self.gridLayout.addWidget(self.label_user,1,\overline{0})
        self.user_st_line =QtGui.QLineEdit(self.tab_sat_tools)
        self.user st line.setObjectName("user st line")
self.gridLayout.addWidget(self.user st line, 1,1)
        self.label_pass =QtGui.QLabel(self.tab sat tools)
        self.label_pass.setObjectName("label_pass")
self.gridLayout.addWidget(self.label_pass,2,0)
        self.pass line =QtGui.QLineEdit(self.tab sat tools)
        self.pass line.setEchoMode (QtGui.QLineEdit.Password)
```

```
self.pass line.setObjectName("pass line")
self.gridLayout.addWidget(self.pass line,2,1)
         self.label_satellites =QtGui.QLabel(self.tab_sat_tools)
         self.label_satellites.setObjectName("label_satellites")
self.gridLayout.addWidget(self.label satellites,3,0)
         self.sat_line =QtGui.QLineEdit(self.tab sat tools)
self.sat_line.setObjectName("sat_line")
self.gridLayout.addWidget(self.sat_line,3,1)
         self.label_api =QtGui.QLabel(self.tab_sat_tools)
         self.label_api.setObjectName("label_api")
self.gridLayout.addWidget(self.label api,4,0)
         self.api_line =QtGui.QLineEdit(self.tab_sat_tools)
self.api_line.setObjectName("api_line")
self.gridLayout.addWidget(self.api line,4,1)
         self.label_date =QtGui.QLabel(self.tab_sat_tools)
         self.label_date.setObjectName("label_date")
self.gridLayout.addWidget(self.label_date,5,0)
         self.date_line =QtGui.QLineEdit(self.tab_sat_tools)
         self.date line.setObjectName("date line")
self.gridLayout.addWidget(self.date line, 5,1)
         self.label_output =QtGui.QLabel(self.tab_sat_tools)
         self.label_output.setObjectName("label_output")
self.gridLayout.addWidget(self.label output,6,0)
         self.out path line =QtGui.QLineEdit(self.tab sat tools)
         self.out_path_line.setObjectName("out_path_line")
self.gridLayout.addWidget(self.out path line,6,1)
         self.out_path_button =QtGui.QToolButton(self.tab_sat_tools)
         self.out path button.setObjectName("out path button")
self.gridLayout.addWidget(self.out path button, 6,2)
        self.label_height =QtGui.QLabel(self.tab_sat_tools)
self.label_height.setObjectName("label_height")
self.gridLayout.addWidget(self.label_height,3,\overline{2})
        self.height_line =QtGui.QLineEdit(self.tab_sat_tools)
self.height_line.setObjectName("height_line")
self.gridLayout.addWidget(self.height line,3,3)
         self.push button1 =QtGui.QPushButton(self.tab sat tools)
         self.push button1.setObjectName("push button1")
self.gridLayout.addWidget(self.push button1,3,4)
         self.text_sat_tools_browser =QtGui.QTextBrowser(self.tab_sat_tools)
         self.text_sat_tools_browser.setObjectName("text_sat_tools_browser")
self.gridLayout.addWidget(self.text sat tools browser,7,0,1,5)
         self.progress bar =QtGui.QProgressBar(self.tab sat tools)
         self.progress_bar.setProperty("value",0)
         self.progress_bar.setObjectName("progress_bar")
self.gridLayout.addWidget(self.progress_bar,8,0,1,5)
         self.file dialog =QtGui.QFileDialog()
         self.settings =QtCore.QSettings()
         self.retranslate ui(self)
         self.create actions()
         self.create menus()
         self.read settings()
         self.settings dialog =SettingsDialog()
# Append tab to the tabbed widget
self.central.addTab(self.tab_sat_tools,"Observation Planning")
         self.user_st_line.setFocus()
# Create a new tab and append it to the tabbed widget
self.central.addTab(self.tab find tracks, "Cross-correlation")
#Creo pool de threads
        self.thread pool =[]
#Asigno flows
self.connect(self.push_button1, QtCore.SIGNAL("clicked()"), self.test)
self.connect(self.out_path_button, QtCore.SIGNAL("clicked()"), self.out_path)
self.connect(self.path_button, QtCore.SIGNAL("clicked()"), self.push_tool)
self.connect(self.push_button, QtCore.SIGNAL("clicked()"), self.run_tracks)
self.connect(self.stop button, QtCore.SIGNAL("clicked()"), self.abort)
defretranslate_ui(self, mainwindow):
mainwindow.setWindowTitle("TFRM Tools")
mainwindow.setWindowIcon(QtGui.QIcon("satellite.ico"))
         self.path button.setText("Path...")
         self.stop_button.setText("STOP")
self.ra.setText("Right Ascension")
```

```
self.dec.setText("Declination")
self.ha.setText("Hour Angle")
        self.push button.setText("RUN")
        self.push_button.setStyleSheet("color: #900; font-weight: bold; font-size:
18px;")
        self.label satellites.setText("Satellites")
        self.label_pass.setText("Password")
self.label_output.setText("Output")
        self.label_api.setText("API")
        self.push_button1.setText("RUN")
        self.push button1.setStyleSheet("color: #900; font-weight: bold; font-size:
18px;")
        self.label date.setText("Date")
        self.label height.setText("Height")
        self.label user.setText("Space-Track User")
        self.out path button.setText("Path...")
    @classmethod
defabort(cls):
returnFalse
defabout(self):
text="<body>"
"<h1>" \
"<b>About TFRM Tools</b> v."+ version + \
"</h1>" \
"<br/>"+"Author: "+ __author__ +"\n"+"<br/>"+\
"<br/>"+"License: "+ __license__ +"<a href =\" http://www.gnu.org/licenses/gpl.txt/\\">"
"http://www.gnu.org/licenses/gpl.txt</a>"+"<br/>"+ \
""
"TFRM Tools is a utility for astronomy calculations." \
" More on <a href =\"http://fridgeonroof.com/\">http://fridgeonroof.com</a>" \
"</body>"
QtGui.QMessageBox.about(self, "About Application", text)
defopen_settings(self):
        self.settings_dialog.exec_()
defcreate actions(self):
        self.exit act =QtGui.QAction(QtGui.QIcon("quit.ico"),'&Quit', self)
        self.exit_act.setShortcut('Ctrl+Q')
self.exit_act.setStatusTip('Exit the application')
self.connect(self.exit act, QtCore.SIGNAL("triggered()"), self.close)
        self.settings act =QtGui.QAction(QtGui.QIcon("tool.ico"),'&Settings...', self)
        self.settings_act.setShortcut('Ctrl+S')
        self.settings act.setStatusTip('Application Settings')
self.connect(self.settings act, QtCore.SIGNAL("triggered()"), self.open settings)
        self.about act =QtGui.QAction(QtGui.QIcon("about.ico"),'&About', self)
        self.about_act.setStatusTip("Show the application's About box")
self.connect(self.about act, QtCore.SIGNAL("triggered()"), self.about)
        self.about_qt =QtGui.QAction(QtGui.QIcon("icon.png"),'&About Qt', self)
        self.about_qt.setStatusTip("Show the Qt library's About box")
self.connect(self.about qt, QtCore.SIGNAL("triggered()"), QtGui.qApp.aboutQt)
defcreate menus(self):
        self.file menu =self.menuBar().addMenu("File")
        self.file menu.addSeparator()
        self.file menu.addAction(self.exit act)
self.menuBar().addSeparator()
        self.settings menu =self.menuBar().addMenu("Settings")
        self.settings menu.addAction(self.settings act)
        self.help_menu =self.menuBar().addMenu("Help")
        self.help menu.addAction(self.about act)
        self.help menu.addAction(self.about qt)
defread settings(self):
settings= QtCore.QSettings("Canals", "Find Tracks")
pos= settings.value("pos", QtCore.QPoint(200,200)).toPoint()
size= settings.value("size", QtCore.QSize(400,400)).toSize()
        path text =settings.value("path").toString()
```

```
dec state =settings.value("dec").toBool()
         ha state =settings.value("ha").toBool()
         ra state =settings.value("ra").toBool()
         user_st_line_text =settings.value("user_st_line").toString()
         pass_line_text =settings.value("pass_line").toString()
         sat line text =settings.value("sat line").toString()
        height_line_text =settings.value("height_line").toString()
api_line_text =settings.value("api_line").toString()
         out_path_line_text =settings.value("out_path_line").toString()
self.resize(size)
self.move(pos)
        self.line edit.setText(path text)
self.dec.setCheckState(dec state)
self.ha.setCheckState(ha state)
self.ra.setCheckState(ra state)
        self.user_st_line.setText(user_st_line_text)
         self.pass_line.setText(pass_line_text)
         self.sat line.setText(sat line text)
         self.height line.setText(height line text)
         self.api line.setText(api line text)
         self.date_line.setText(str(ephem.date(dt.datetime.now())))
         self.out_path_line.setText(out_path_line_text)
defwrite settings(self):
settings= QtCore.QSettings("Canals", "Find Tracks")
settings.setValue("dec", self.dec.checkState())
settings.setValue("ha", self.ha.checkState())
settings.setValue("ra", self.ra.checkState())
settings.setValue("user_st_line", self.user_st_line.text())
settings.setValue("pass_line", self.pass_line.text())
settings.setValue("sat_line", self.sat_line.text())
settings.setValue("height_line", self.height_line.text())
settings.setValue("api_line", self.api_line.text())
settings.setValue("out path line", self.out path line.text())
defcloseEvent(self, event):
msgbox= QtGui.QMessageBox()
msgbox.setWindowTitle("Quit")
msgbox.setText("Are you sure?")
msgbox.setWindowIcon(QtGui.QIcon("quit.ico"))
msgbox.setStandardButtons(QtGui.QMessageBox.Yes | QtGui.QMessageBox.No)
ret= msgbox.exec ()
if ret == QtGui.QMessageBox.Yes:
             self.write_settings()
event.accept()
else:
event.ignore()
defpush tool(self):
        p = self.file dialog.getOpenFileName(None, "Select File", self.line edit.text())
             self.line edit.setText(p)
defout path(self):
        p =str(self.file dialog.getExistingDirectory(self, "Select Directory"))
             self.out path line.setText(p)
defdb connection(self):
path= self.line_edit.text()
if path[-4:]==".txt":
db.init(':memory:')
db.connect()
try:
with open(str(path)+'.csv','wb')as pc:
with open(str(path),'rb')as cand txt:
                          i =0
                          j =0
for line in cand txt:
columns= line.split()
if"/"in line:
date= line[6:10]+"-"+ line[3:5]+"-"+ line[0:2]+" "
tm="0"+ line[11:18]
```

```
datetm=[date + tm]
data=[]
else:
data= columns[0:6]
                                j +=1
if datetm and data:
row=[str(j)]+[str(i)]+ datetm + data
                                a =csv.writer(pc, delimiter=';')
a.writerow(row)
               load csv(Points, str(path)+'.csv', delimiter=';', has header=False)
os.remove(str(path)+'.csv')
except IOError:
                b ="<b>File </b> "+str(path)+" <b>not found</b>"
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), b)
os.remove(str(path)+'.csv')
returnFalse
except NameError:
               b ="<b>File </b> "+str(path)+" <b>invalid format</b>"
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), b)
os.remove(str(path)+'.csv')
returnFalse
            Finaltracks.create_table()
            a ="No *.txt file selected, please select correct extension file.\n"
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
returnFalse
defadd(self, text):
""" Add item to list widget """
        self.text_sat_tools_browser.append(text)
deffill_progress_bar(self, step):
"""Fills the progress bar"""
        self.progress_bar.setValue(step +1)
deftest(self):
# print self.thread pool
        self.text_sat_tools_browser.clear()
        self.progress bar.reset()
        self.thread_pool.append(GenericThread(self.execute))
self.disconnect(self, QtCore.SIGNAL("add(QString)"), self.add)
self.connect(self, QtCore.SIGNAL("add(QString)"), self.add)
self.mysignal.connect(self.fill_progress_bar)
        self.thread_pool[len(self.thread_pool)-1].start()
defexecute(self):
cadena='Connecting.....wait please\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
contador no visible =0
       contador navstar =0
contador_errores =0
if self.sat_line.text() == "":
            fichero datos ='AllSats-'+dt.datetime.now().strftime('%Y%m%d%H%M')+'.txt'
else:
            fichero_datos =str(self.sat_line.text())+'-'+
dt.datetime.now().strftime('%Y%m%d%H%M')+'.txt'
cj= cookielib.CookieJar()
website= urllib2.build opener(urllib2.HTTPCookieProcessor(cj))
       login data =urllib.urlencode({'identity': self.user st line.text(), 'password':
self.pass line.text()})
trv:
resp= website.open('https://www.space-track.org/ajaxauth/login', login_data)
if resp.read()!='""':
cadena='Login fail. Review username and password\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
return
except:
cadena='No internet connection\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
return
resp= website.open(str(self.api line.text()))
except:
```

```
cadena='Error in API. Review API and execute again.\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
return
reader= csv.reader(resp)
website.close()
cadena='Connection successful!!!\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
fichero= open(os.path.join(str(self.out path line.text()), fichero datos),'w')
tfrm= ephem.Observer()
        tfrm.lon = '0.729354'
        tfrm.lat ='42.051599'
        tfrm.elevation = 1572
        11 =[]
        12 =[]
        13 =[]
        sat_list =[]
for index, row in enumerate(reader):
11.append(row[23])
12.append(row[24])
13.append(row[25])
        satelites totales = index
        self.progress bar.setMaximum(satelites totales +1)
for i in range(1, satelites_totales +1):
            tfrm.date =str(self.date line.text())
self.mysignal.emit(i)
try:
                sat list = ephem.readtle(str(l1[i]), str(l2[i]), str(l3[i]))
info= tfrm.next_pass(sat_list)
except ValueError:
                contador_errores +=1
            14 =11[i]
            15 =12[i]
            16 =13[i]
            rise time =info[0]
            azimut_rise_time =info[1]
            transit_time =info[2]
            altura transit time =info[3]
            set time =info[4]
            azimut_set_time =info[5]
            h = []
            a =[]
azi=[]
if rise time:
while rise_time <= set_time:</pre>
                    tfrm.date = rise time
                    sat list.compute(tfrm)
h.append(tfrm.date)
a.append(math.degrees(sat_list.alt))
azi.append(math.degrees(sat list.az))
                    rise time =ephem.date(rise time +60.0* ephem.second)
if a:
ind= a.index(max(a))
                h1 = h[ind]
                azi1 =azi[ind]
if 14.find(self.sat_line.text())>-land max(a)> float(self.height line.text()):
fichero.write(str(1\overline{4})+'\n'+ str(15)+'\n'+ str(16)+'\n') fichero.write('Date/Time (UTC) Maxima Altura
                                                           Azimut'+'\n')
                    fichero.write(str(h1)+'.....'+ str(max(a))+'.....'+
str(azi1)+'\n')
fichero.write('======='+'\n')
                    contador_navstar +=1
elif self.sat line.text() ==-land max(a) > float(self.height line.text()):
```

```
fichero.write(str(14)+'n'+ str(15)+'n'+ str(16)+'n')
fichero.write('Date/Time (UTC)
                                        Máxima Altura
                                                             Azimut'+'\n')
                     fichero.write(str(h1)+'....'+ str(max(a))+'....'+
str(azi1)+' n')
fichero.write('====
contador navstar+=1
else:
contador no visible+=1
fichero.close()
\# cadena = '\n'
cadena='Total satellites analyzed = '+ str(satelites totales)+'\n'
cadena+='Not visible satellites from the TFRM = '+ str(contador no visible)+'\n'
if self.sat_line.text() == "":
cadena+='Visible satellites from the TFRM'+' = '+ \
str(contador_navstar)+'\n'
else:
cadena+='Visible satellites from the TFRM of the type '+ self.sat line.text()+' = '+ \
str(contador navstar)+'\n'
{\tt cadena+='Miscalculations\ due\ to\ tle\ errors\ =\ '+\ str(contador\_errores)+'\setminus n'}
cadena+='Great job! Have a nice day!'+'\n'
self.emit(QtCore.SIGNAL('add(QString)'), cadena)
return
defadd_tracks_to_textbox(self, text):
""" Add item to list widget """
        self.text browser.append(text)
defrun tracks(self):
        self.thread_pool.append(GenericThread(self.push_run))
self.disconnect(self, QtCore.SIGNAL("add tracks to textbox(QString)"),
self.add_tracks_to_textbox)
self.connect(self, QtCore.SIGNAL("add tracks to textbox(QString)"),
self.add_tracks_to_textbox)
        self.thread pool[len(self.thread pool)-1].start()
defpush_run(self):
        a ='<br><b>Calculating, please wait...</b><br>'
        TrackFinder.total tracks =[]
if self.db connection()isFalse:
return
candidates= Points.select()
        self.widget.get_figure().clear()
self.widget_2.get_figure().clear()
subplot= self.widget.get_figure().add_subplot(111)
if self.dec.isChecked() and self.ha.isChecked() andnot self.ra.isChecked():
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
for candidate in candidates:
subplot.plot(candidate.ha, candidate.dec,"or")
            subplot.set_title("Initial Tracks")
subplot.set_xlabel("HA")
            subplot.set ylabel("DEC")
self.widget.draw()
            f =TrackFinder()
            f.filter_points_by_ha_dec()
            subplot1 = self.widget_2.get_figure().add_subplot(111)
for p in TrackFinder.total_tracks:
                ha_trys =[]
                dec trys =[]
for pp in p:
Finaltracks.select().where(Finaltracks.id == pp.id).get()
                     ha trys.append(pp.ha)
                     dec_trys.append(pp.dec)
subplot1.plot(ha_trys, dec_trys)
            subplot1.set title("Final Tracks")
            subplot1.set xlabel("HA")
            subplot1.set_ylabel("DEC")
self.widget_2.draw()
            a ='<b>Final Tracks:</b><br>'
            track number =0
for track in TrackFinder.total_tracks:
                track number +=1
                a +="<b>Track %s: </b><br>"% track number
for point in track:
```

```
a +="id: %s, ha: %s, dec: %s, ra: %s <br>"%(point.id, point.ha,
point.dec, point.ra)
                a +="<br>"
            a +="<b>Total tracks found --> %s</b>"%len(TrackFinder.total_tracks)
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
elif self.dec.isChecked() and self.ra.isChecked() andnot self.ha.isChecked():
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
for candidate in candidates:
subplot.plot(candidate.ra, candidate.dec,"or")
            subplot.set_title("Initial Tracks")
            subplot.set_xlabel("RA")
            subplot.set ylabel("DEC")
self.widget.draw()
            f =TrackFinder()
            f.filter_points_by_ra_dec()
            subplot1 = self.widget_2.get_figure().add_subplot(111)
for p in TrackFinder.total tracks:
                ra trys =[]
                dec trys =[]
for pp in p:
Finaltracks.select().where(Finaltracks.id == pp.id).get()
                    ra trys.append(pp.ra)
                    dec trys.append(pp.dec)
subplot1.plot(ra_trys, dec_trys)
            subplot1.set_title("Final Tracks")
            subplot1.set_xlabel("RA")
            subplot1.set_ylabel("DEC")
self.widget_2.draw()
            a ='<b>Final Tracks:</b><br>'
            track number =0
for track in TrackFinder.total tracks:
                track_number +=1
                a +="<b>Track %s: </b><br>"% track number
for point in track:
                    a +="id: %s, ha: %s, dec: %s, ra: %s <br>"%(point.id, point.ha,
point.dec, point.ra)
                a +="<br>"
            a +="<b>Total tracks found --> %s</b>"%len(TrackFinder.total tracks)
self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
elif self.ra.isChecked() and self.ha.isChecked() andnot self.dec.isChecked():
            a ='<h3 style=\"color:#FF0000\">Invalid selection</h3>'
return self.emit(QtCore.SIGNAL('add_tracks_to_textbox(QString)'), a)
            a = ' < h3  style=\"color:#FF0000\">Select 2 of the checkboxes</h3>'
return self.emit(QtCore.SIGNAL('add tracks to textbox(QString)'), a)
classGenericThread(QtCore.QThread):
def init (self, function):
super(GenericThread, self). init ()
        self.function = function
def del
         (self):
self.wait()
defrum(self):
self.function()
classSettingsDialog(QtGui.QDialog):
def init
          (self):
super (SettingsDialog, self). init
        self.db_line =QtGui.QLineEdit()
        self.host line =QtGui.QLineEdit()
        self.port_line =QtGui.QLineEdit()
        self.user_line =QtGui.QLineEdit()
self.pass_line =QtGui.QLineEdit()
        \verb|self.pass_line.setEchoMode(QtGui.QLineEdit.Password)|\\
db label =QtGui.QLabel("&Database:")
db label.setBuddy(self.db line)
        host label =QtGui.QLabel("&Host:")
```

```
host label.setBuddy(self.host line)
         port_label =QtGui.QLabel("&Port:")
port_label.setBuddy(self.port_line)
user_label =QtGui.QLabel("&User:")
user label.setBuddy(self.user line)
         pass label =QtGui.QLabel("&Password:")
         pass label.setBuddy(self.pass line)
         self.connection box =QtGui.QGroupBox("Test connection")
         self.connection_log =QtGui.QTextBrowser()
         self.conn button =QtGui.QPushButton(self.tr("&Test Connection"))
         self.conn button.setDefault(True)
         self.button box = QtGui.QDialogButtonBox(QtGui.QDialogButtonBox.Ok |
QtGui.QDialogButtonBox.Cancel)
         self.button box.addButton(self.conn button, QtGui.QDialogButtonBox.ActionRole)
self.connect(self.conn button, QtCore.SIGNAL("clicked()"), self.on push conn button)
         self.button box.accepted.connect(self.ok event)
         self.button box.rejected.connect(self.reject)
         locations layout =QtGui.QVBoxLayout()
         locations layout.addWidget(self.connection log)
         self.connection box.setLayout(locations layout)
         main layout =QtGui.QGridLayout()
         main_layout.addWidget(db_label,0,0)
         main layout.addWidget(self.db line,0,1)
         main layout.addWidget(host label, 1, 0)
         main_layout.addWidget(self.host line,1,1)
         main_layout.addWidget(port_label,2,0)
         main_layout.addWidget(self.port_line,2,1)
         main layout.addWidget(user label,3,0)
         main layout.addWidget(self.user line,3,1)
         main_layout.addWidget(pass_label,4,0)
         main layout.addWidget(self.pass line,4,1)
         main_layout.addWidget(self.connection_box,5,0,1,2)
         main layout.addWidget(self.button box, 6, 0, 1, 3)
self.setLayout(main layout)
self.setWindowTitle("Application Settings")
self.setWindowIcon(QtGui.QIcon("tool.ico"))
         self.read settings()
# self.resize(650, 400)
defread settings(self):
settings= QtCore.QSettings("Canals", "App Settings")
pos= settings.value("pos", QtCore.QPoint(200,200)).toPoint()
size= settings.value("size", QtCore.QSize(400,400)).toSize()
         database_text =settings.value("database").toString()
         host_text =settings.value("host").toString()
         port_text =settings.value("port").toString()
user_text =settings.value("user").toString()
         pass text =settings.value("pass").toString()
self.resize(size)
self.move(pos)
         self.db line.setText(database text)
         self.host_line.setText(host_text)
         self.port_line.setText(port_text)
self.user_line.setText(user_text)
         self.pass line.setText(pass text)
defwrite settings(self):
settings= QtCore.QSettings("Canals", "App Settings")
settings.setValue("pos", self.pos())
settings.setValue("size", self.size())
settings.setValue("database", self.db line.text())
settings.setValue("host", self.host_line.text())
settings.setValue("port", self.port_line.text())
settings.setValue("user", self.user_line.text())
settings.setValue("pass", self.pass_line.text())
defok event(self):
        self.write_settings()
self.accept()
defon push conn button (self):
try:
database= str(self.db line.text())
host= str(self.host_line.text())
port= int(self.port line.text())
user= str(self.user line.text())
passwd= str(self.pass line.text())
```

```
datb= pw.MySQLDatabase(database, host=host, port=port, user=user, passwd=passwd)
except ValueError:
        a =str(dt.datetime.now().strftime("%H:%M:%S"))+ \
self.connection_log.append(a)
try:
if datb.get conn():
datb.connect()
           self.connection_log.append(a)
except(pw.OperationalError, TypeError):
        a =str(dt.datetime.now().strftime("%H:%M:%S"))+ \
self.connection log.append(a)
if __name__ =="__main__":
app= QtGui.QApplication(sys.argv)
ui= Mainwindow()
ui.show()
# sys.exit(app.exec_())
app.exec_()
```

APPENDIX 3. Punto.py

```
# -*- coding: utf-8 -*-
__license__ ='GPL v.3'
__author__ ="Lluis Canals Ros <lluiscanalsros@gmail.com>"
__version__ ='0.4'
classPunto(object):
def__init__(self, ide, ha, dec, ra):
    self.id = ide
          self.ha = ha
self.dec = dec
          self.ra = ra
          self.evaluated =False
def__str__(self):
    return str(self.id)+": ["+ str(self.ha)+", "+ str(self.dec)+", "+
    str(self.ra)+", "+ str(self.evaluated)+"]"
def__cmp__(self, other):
if self.id < other.id:</pre>
return-1
elif self.id > other.id:
return1
else:
return0
def__hash__(self):
return hash (self.id)
```

APPENDIX 4. global_cand.txt

This is a summary of the complete generated file.

```
25/05/2011 0:32:00 40688.02222358 10.000
    1705.651000 476.403000 10.040550 14.544292 -6.400498 10.106000
    305.138000 1505.622000 10.113924 14.470918 -4.883638 11.689000
    1597.405000 1955.878000 10.147307 14.437535 -6.270643 9.745000
    1627.285000 1980.679000 10.149120 14.435722 -6.302571 11.047000
    1662.495000 2024.630000 10.152322 14.432520 -6.340034 11.531000
    1681.762000 2034.766000 10.153069 14.431773 -6.360683 10.812000
    341.628000 2323.090000 10.172837 14.412005 -4.914126 12.899000
    1292.624000 2731.529000 10.203075 14.381767 -5.933385 13.005000
10
    1663.040000 3545.033000 10.262159 14.322683 -6.321625 11.372000
    1525.295000 3297.277000 10.244133 14.340709 -6.176698 11.031000
11
12
    3881.241000 3242.968000 10.242478 14.342364 -8.714768 13.944000
    25/05/2011 0:33:30 40688.02326558 10.000
    1698.655000 504.336000 10.042571 14.567348 -6.393016 10.266000
    120.073000 862.179000 10.067508 14.542411 -4.690949 10.343000
    318.903000 1505.774000 10.113954 14.495965 -4.898757 11.725000
17
    1597.607000 1956.082000 10.147328 14.462591 -6.271110 9.723000
18
    1627.672000 1981.059000 10.149154 14.460765 -6.303234 11.059000
    1662.300000 2024.677000 10.152331 14.457588 -6.340072 11.524000
19
20
    1681.810000 2034.985000 10.153091 14.456828 -6.360980 10.799000
21
    418.880000 2333.875000 10.173679 14.436240 -4.997428 12.918000
    1301.061000 2732.478000 10.203156 14.406763 -5.942698 13.144000
22
    4054.536000 2907.732000 10.218317 14.391602 -8.906211 14.745000
    3885.795000 3587.844000 10.267530 14.342389 -8.714868 13.857000
24
    1529.290000 3252.607000 10.240919 14.369000 -6.181848 11.031000
26
    1663.038000 3545.125000 10.262175 14.347744 -6.321881 11.376000
    1720.024000 3091.364000 10.229451 14.380468 -6.389476 13.564000
    25/05/2011 0:37:00 40688.02569580 10.000
    1682.203000 570.017000 10.047307 14.621097 -6.375458 10.267000
29
    219.310000 868.284000 10.068006 14.600398 -4.798298 10.326000
    144.962000 951.948000 10.073983 14.594421 -4.717563 11.971000
31
    351.188000 1506.028000 10.113997 14.554407 -4.934142 11.706000
    1597.862000 1956.719000 10.147386 14.521018 -6.272000 9.740000
    1628.431000 1982.012000 10.149236 14.519168 -6.304663 11.051000
34
    1661.722000 2024.839000 10.152355 14.516049 -6.340068 11.531000
    1681.788000 2035.689000 10.153154 14.515250 -6.361569 10.810000
36
    1323.101000 2747.134000 10.204244 14.464160 -5.966880 13.059000
    1663.133000 3545.539000 10.262211 14.406193 -6.322584 11.378000
38
39
    1281.560000 4058.184000 10.298814 14.369590 -5.904117 15.366000
40
    1914.514000 3120.048000 10.231715 14.436689 -6.599233 13.351000
41
    1538.863000 3150.091000 10.233534 14.434870 -6.194144 11.008000
    25/05/2011 0:39:00 40688.02708470 10.000
    13.094000 223.908000 10.021572 14.680257 -4.582427 12.728000
43
    1672.702000 607.731000 10.050027 14.651802 -6.365376 10.348000
    241.631000 975.221000 10.075724 14.626105 -4.821727 11.918000
45
    1791.955000 1351.765000 10.103824 14.598005 -6.487558 14.960000
    369.757000 1506.198000 10.114032 14.587797 -4.954499 11.684000
    225.869000 1931.057000 10.144530 14.557299 -4.795208 11.762000
```

REFERENCES

- Fors, O., Núñez, J., Muiños, J.L., et. al., Telescope Fabra ROA Montsec: A New Robotic Wide Field Baker-Nunn Facility, Publications of the Astronomical Society of the Pacific, 2013.http://en.wikipedia.org/wiki/Low_Earth_orbit#mediaviewer/File:Orbitalaltitudes .jpg
- http://www.esa.int/var/esa/storage/images/esa_multimedia/images/2013/04/impact _crater_size_4_mm_on_solar_cell_retrieved_from_space/12625511-1-eng-GB/Impact_crater_size_4_mm_on_solar_cell_retrieved_from_space.jpg
- 3. Klinkrad, H., DISCOS ESA's database and information system characterizing objects in space, Advances in Space Research, Vol. 11, pp. 43-52, 1991.
- 4. https://www.space-track.org
- 5. https://www.python.org/
- 6. http://www.riverbankcomputing.co.uk/software/pyqt/intro
- 7. http://matplotlib.org/
- 8. http://peewee.readthedocs.org/en/latest/
- 9. http://docs.scipy.org/doc/scipy/reference/
- 10. http://rhodesmill.org/pyephem/





Máster Universitario en Ingeniería Geodésica y Cartografía.
Universidad Politécnica de Madrid.

INTRODUCCIÓN AL MANEJO DEL SOFTWARE GOA EN PPP. APLICACIÓN PRÁCTICA A LA GEODINÁMICA DE LA ISLA DE ALBORÁN

Trabajo realizado por Ángel Cibeira Urtiaga

Directores del Trabajo:

Dr. Abelardo Bethencourt Fernández

Dr. Jorge Gárate Pasquín

Universidad Politécnica de Madrid

Real Observatorio de la Armada

Madrid, Enero de 2015.

Agradecimientos.

Este proyecto ha supuesto un gran número de horas de trabajo, y sin el apoyo incondicional de mi familia, en especial de mi mujer, no hubiese sido posible terminarlo. Muchas gracias Patricia por tu respaldo y comprensión. Sirvan también estas líneas para expresar mi más sentido agradecimiento a todas las personas que de una u otra manera me han ayudado, en especial a mi tutor Abelardo Bethencourt por todas las facilidades que me ha dado, y a mi cotutor, el Capitán de Navío Gárate por todo el tiempo dedicado, más allá del profesionalmente exigible.

Especial mención al Capitán de Corbeta Héctor Esteban, que me ayudó y guió en mis primeros pasos, y sin cuyos consejos este trabajo se hubiese hecho muy cuesta arriba. Agradecer también al Capitán de Fragata Antonio Pazos, jefe de la sección de Geofísica del ROA, a la cual pertenezco, y que me dio una gran libertad de trabajo y colaboración. Quisiera también extender este agradecimiento a mis compañeros de la sección de Geofísica y en especial a los miembros del servicio de satélites por su comprensión y paciencia conmigo, y en general al ROA como institución, que puso al alcance de mi manos todos los medios disponibles para la consecución de este trabajo.

Finalmente, agradecer a los profesores de la UPM, a cuyas clases he tenido el privilegio de asistir, las horas de enseñanza y la rápida y atenta ayuda que siempre me han dado.

Índice:

1. Introducción	121
1.1 Objetivos	122
2. Sistemas GNSS	123
2.1 El sistema GPS como ejemplo típico de constelación de navegación	
2.1.1 Segmento especial	124
2.1.2 Segmento de Control	124
2.1.3 Segmento usuario	125
2.2 Observables GPS	126
2.2.1 Pseudodistancia	127
2.2.1.1 Modelo matemático	128
2.2.2 Fase de la portadora	128
2.2.2.1 Modelo matemático	128
2.3 Errores	132
2.3.1 Errores debidos a las efemérides y estados del satélites	-
2.3.2 Errores debidos a la propagación de la señal a t atmósfera	
2.3.3 Errores debidos al receptor y su entorno	136
3.	
Geodesia y Aplicaciones	137
3.1 Simples diferencias	139
3.2 Dobles diferencias	141
3.3 Triples diferencias	143
3.4 Órbitas precisas	144

4.

Evolución PPP	145
4.1 PPP	145
4.2 Método	145
4.2.1 Marco de referencia	146
4.2.2 Ecuaciones de observación	149
4.2.3 Modelo de ajuste	150
4.2.4 Procedimiento de Ajuste	150
4.2.5 Modelos de corrección aplicados al posicionamiento preciso	
4.2.5.1 Efectos derivados de la actitud del satélite	153
4.2.5.1.1 Offsets de la antena del satélite	154
4.2.5.1.2 Efecto wind up	154
4.2.5.2 Efectos derivados del emplazamiento	154
4.2.5.2.1 Carga de marea terrestre	155
4.2.5.2.2 Marea polar	155
4.2.5.2.3 Carga de marea oceánica	155
4.2.5.2.4 Parámetros de orientación de la Tierra	156
4.2.5.3 Compatibilidades	157
4.3 Estrategia de cálculo	157
4.3.1 Convergencia de la solución PPP	158
5.	
GIPSYOASIS	159
5.1 Funcionamiento en red	159
5.1.1 Ninja	159
5.1.2 Merg_qm	160
5.1.3 Genoi	162
5.1.4 Qregres	164
5.1.5 El filtrado	164
5.1.6 Smapper	165

5.1.7 Posfit	165
5.1.8 Postbreak	166
5.1.9 Edtpnt2	166
5.1.10 Stacov	167
5.3 Gipsy en PPP. Gd2p.pl	168
5.3.1 Ficheros de estación	168
5.3.2 Preparación para el procesado	169
5.3.3 Gd2p.pl	170
6.	
Aplicación práctica a la geodinámica de la Isla de Alborán	171
6.1 Descripción de la región	171
6.2 Descripción de los equipos	173
6.3 Configuración del receptor	175
6.4 Procesado de los datos y conclusiones	176
6.5 Trabajos futuros	180
Anexo I:	
Script crea_angel	183
Anexo II:	
script solo_angel	185
Anexo III:	
script solo_new_angel	187
AnexolV: gipsy2gd	188
AnexoV: gipsy2xyz	189
Bibliografía	191

Capítulo 1

Introducción

La Geodesia es la ciencia que estudia el tamaño y la forma de la Tierra. Desde sus inicios, la mayor dificultad a la que se enfrentaba era la limitación impuesta por las observaciones, que ya fuesen observaciones terrestres (como nivelaciones o triangulaciones) o astronómicas (determinación de posiciones y azimutes), se realizaban sobre la superficie del planeta. El lanzamiento del primer satélite artificial en 1957 (Sputnik, perteneciente a la extinta Unión Soviética) marcó un punto de inflexión, apareciendo la posibilidad de montar equipos para la observación de la Tierra a bordo de estos vehículos espaciales. De esta manera surge lo que se conoce como Geodesia por satélite, i.e. la observación de la Tierra para determinar su tamaño y forma empleando satélites artificiales sometidos al efecto del campo gravitatorio terrestre, lo que permite también el estudio del campo gravitatorio externo de la Tierra y su variación en el tiempo.

A medida que se desarrollaban y mejoraban las capacidades de equipos y satélites, las aplicaciones de la Geodesia Espacial fueron aumentando. Una de las mayores contribuciones de esta ciencia a la sociedad (tal vez no la mayor pero sí la que más repercusión ha tenido) ha sido la del posicionamiento por satélite. El problema de posicionamiento en un sistema de referencia global se hacía mediante cálculos derivados de observaciones astronómicas. El posicionamiento por satélite fue llamado primer sistema de el TRANSIT/NAVSAT/NNSS, empleado por la U.S. Navy para la ubicación de barcos y submarinos, de cara al lanzamiento de misiles balísticos. Éste sistema, con sus limitaciones (por ejemplo no se podía emplear en navegación aérea) estuvo plenamente operativo en la primera mitad de la década de los sesenta, permitiéndose en 1967 su uso civil, y estando operativo durante 33 años. Este sistema llegó a emplearse en geodesia.

En 1973 el sistema de posicionamiento nace por satélite NAVSTAR/GPS, desarrollado a partir de los proyectos TIMATION y 621B, creado en la década de los 60 con la idea de construir un sistema de posicionamiento pasivo global que emplease medidas de distancias. La implantación definitiva del sistema finalizó en 1993, siendo declarado operativo el 17 de julio de 1995. El sistema, en su inicio lo componían 24 satélites del bloque I/!!A, divididos en 6 planos orbitales a razón de 4 por plano. El sistema a su vez se compone de tres segmentos (espacial, control y usuario) que serán descritos en capítulos posteriores.

Además de las ya reconocidas capacidades de posicionamiento y navegación, un tratamiento interferométrico de la señal GPS, da como resultado un posicionamiento relativo de gran precisión entre estaciones que

simultáneamente reciban la señal. Ésta capacidad es una herramienta de gran utilidad para estudios geodésicos y geofísicos. El principal inconveniente que presenta ésta forma de trabajo, es la necesidad de conocer con gran exactitud la posición de uno de los receptores, además de emplear varios de ellos. El problema se puede resolver empleando un único receptor bifrecuencia, que emplea dos frecuencias para posicionarse, viéndose mermada la precisión. A este método se le conoce como PPP (Precise Point Positioning).

Para poder procesar con fines geodésicos los datos procedentes de constelaciones GNSS (Global Navigation Satellite System), se han desarrollado una serie de softwares, como son Bernesse, creado en la Universidad de Berna, Gamit, creado en el MIT (Massachusets Institute of Technology) o el GIPSY-OASIS, desarrollado por el JPL (Jet Propulsion Laboratory) de la NASA, y es el que se describirá y empleará en el presente trabajo, y que implementará soluciones PPP para una serie temporal de una estación GPS situada en la isla de Alborán.

1.1 OBJETIVOS

El objetivo principal de este trabajo es la descripción general del funcionamiento del software de procesamiento científico de datos GNNS GIPSY-OASIS, qué parámetros y variables emplea etc. Así mismo se describirá el funcionamiento de la técnica PPP y de las precisiones que se consiguen con ella. Como ejemplo típico de sistema de posicionamiento por satélite se utilizara la constelación GPS. Para ello, se describirán los distintos segmentos y observables que utiliza este sistema, así como de las posibles estrategias de procesado. Por último, se realizará una aplicación práctica empleando el software GIPSY-OASIS, empleando los datos de la estación permanente GPS instalada por el Real Observatorio de la Armada en la isla de Alborán, en el mar con el mismo nombre, a 100 millas náuticas al Oeste del Estrecho de Gibraltar.

Capítulo 2. Sistemas GNSS.

2.1 El sistema GPS como ejemplo típico de constelación de satélites de navegación.

El GPS es un sistema de posicionamiento por satélite en el que la posición se calcula a través de las señales emitidas por los vehículos orbitales. Su desarrollo comienza en la década de 1970, siendo llevado a cabo por el Departamento de Defensa de los Estados Unidos. Las primeras aplicaciones del sistema para la navegación se emplean en la década de 1980, estando plenamente operativo a mediados de la década de 1990.

Para su funcionamiento, se emplea la medida de la distancia entre el receptor cuyas coordenadas quieren ser determinadas, y las de un mínimo de cuatro satélites de la constelación, cuyas posiciones se conocen con gran precisión. Se necesitan cuatro, y no tres satélites para la determinación de la posición, ya que el estado del reloj del receptor es una incógnita más a introducir en la ecuación de posicionamiento a resolver.

El sistema consta de tres partes principales: el segmento espacial, constituido por una serie de satélites que transmiten señales en frecuencias moduladas a partir de relojes atómicos montados a bordo; el segmento de control, consistente en un grupo de estaciones de seguimiento terrestres; y el segmento usuario, formado por los receptores pasivos, disponibles para cualquier usuario. A continuación se describen con más profundidad cada uno de los segmentos anteriormente mencionados:

2.1.1 SEGMENTO ESPACIAL:

La constelación está definida mediante 24 satélites operativos, a razón de 4 por plano orbital, en un total de 6 planos orbitales, denominados por letras desde la A hasta la F. Las órbitas que recorren los satélites son cuasicirculares, con un semieje mayor de 26.600 Km. Y 55 grados de inclinación, con un periodo de revolución de 12 horas sidéreas. En la actualidad existen en órbita satélites de los bloques IIA, IIR, IIR-M y IIF.

En un principio se equiparon con relojes de Cesio y Rubidio, pero debido al pobre resultado que daban los de Cesio, los bloques IIR y los IIR-M sólo montan relojes basados en patrones de Rubidio. Cada satélite va equipado con varios relojes, aunque en cada satélite hay un único reloj en servicio, y una vez que finaliza su vida útil o sufre alguna avería, otro entra en funcionamiento. Los relojes generan una frecuencia fundamental de 10.23 MHz, a partir de la cual

se crean las dos frecuencias portadoras, a saber, L1 de 1575,42 MHz y L2 de 1277,6 MHz. El emplear dos frecuencias de transmisión tiene como ventaja la determinación del retardo ionosférico, por comparación del retraso de cada frecuencia. Las portadoras son moduladas por dos códigos: C/A (sólo en L1, aunque en los últimos años, los nuevos satélites lanzados ya están habilitados para transmitir el código C/A a través de la frecuencia L2) compuesto por un grupo de 1023 dígitos binarios, con una duración de un milisegundo; y P (tanto en L1 como en L2) compuesto también por una secuencia de dígitos binarios, más larga que la del código anterior, con una periodicidad de 7 días, que pertenece a un código de 266 días de duración. El código C/A (clear access) es de uso libre, mientras que al P (Precise) sólo tienen acceso los usuarios autorizados.

Los satélites del bloque IIF incluyen una nueva frecuencia de transmisión, la L5, a 1176,45 MHz. Esta frecuencia cuenta con un ancho de banda mayor, lo que permite una modulación más alta y también una mayor transmisión en los extremos del ancho de banda, permitiendo una mayor precisión en presencia de ruido y efecto multipath. Además, es capaz de estimar el retardo ionosférico con mayor exactitud, ya que se un único satélite transmite en tres frecuencias diferentes.

El fin de estos códigos es hallar la distancia que separa el satélite del receptor, midiendo el tiempo de vuelo de la señal. Si se simultanean varias de estas mediciones a satélites distintos de posiciones conocidas (transmitidas en el mensaje de navegación) es posible hallar la posición del receptor. En el mensaje de navegación anteriormente mencionado no sólo se transmite la posición del satélite. Además se transmite información del estado del reloj, efemérides del satélite, estado de funcionamiento del satélite y modelos de correcciones para mitigar el efecto ionosférico en la propagación de la señal.

La constelación actual habilita a un observador situado en cualquier punto de la Tierra a tener un número entre 6 y 11 satélites sobre el horizonte. Existen distintas formas de identificar a los satélites:

- Por el Número de vehículo (NAVSTAR SVN), que indica el orden con el que fueron puestos en órbita.
- Por el plano orbital y su posición en dicho plano.
- Por el número de catálogo de la NASA.
- Por el año de lanzamiento y número de lanzamiento dentro de ese año, junto con una letra que identifica el bloque al que pertenece.
- Por el número IRON (Integer Range Operation Number).
- Por el ruido pseudo aleatorio, característico de cada satélite (PRN).

2.1.2 SEGMENTO DE CONTROL.

Está formado por una serie de estaciones de seguimiento y control estratégicamente distribuidas, de manera que se pueda hacer un seguimiento continuo de la constelación. El segmento de control tiene como principal cometido el funcionamiento y operatividad del sistema GPS. Se compone de una estación principal, situada en Schriever, en la base aérea Falcon en Colorado Springs, siendo este el lugar donde se recopila y procesa la información recogida por otras cuatro estaciones secundarias, emplazadas en Hawaii, Isla Ascensión, Diego García y Kwajalein. Los datos recopilados son procesados con objeto de calcular las efemérides de los satélites y los estados de los relojes que llevan a bordo, calculándose cada 15 minutos. Además en Colorado Springs, también se registran datos meteorológicos con el fin de realizar modelos ionosféricos que contribuyan a aumentar la precisión de los resultados.

Los datos obtenidos son transmitidos a los satélites una vez al día, empleando antenas de transmisión en tierra. Las estaciones capaces de realizar ésta transmisión de datos son : Isla Ascensión, Diego García y Kwajalein. Desde Colorado Springs (también conocida como Master Control Station) se controlan las maniobras orbitales de los satélites, configuración del sistema y mantenimientos.

2.1.3 SEGMENTO USUARIO.

Se conoce como segmento usuario a todo receptor de señales GPS, empleado en el cálculo de la posición dónde se encuentra o el tiempo, y que puede ser utilizado en navegación. Son equipos pasivos (no emiten), y se componen de receptor, antena y el software necesario para el procesado de datos.

Los receptores deben de tener la capacidad de seguir un mínimo de 4 satélites, para ser capaces de conseguir navegación 3D. Se pueden agrupar en dos clasificaciones: según la arquitectura de construcción o según el tipo de observable que registran. Los primeros pueden ser multicanales paralelos, secuenciales, asociando un canal para cada satélite y multiplexados. Los segundos, pueden registrar pseudodistancia con código C/A, fase de la portadora con código C/A y fase de la portadora con código P.

2.2 Observables GPS.

Por observables GPS nos referimos a los datos con los que trabaja el sistema. La base para hallar estos observables es la medida del tiempo de vuelo de una señal o de diferencias de fase, obteniéndose a partir de comparación entre la señal recibida en el receptor y la que él mismo genera. Estas observables son dos: la pseudodistancia y la fase de la portadora. La combinación de estas dos observables presenta ventajas de cara al empleo geodésico de los sistemas de posicionamiento por satélite.

2.2.1 PSEUDODISTANCIA.

La pseudodistancia es la medida de la distancia (incluyendo los errores que se describirán más adelante) entre el satélite y la antena del receptor en la época de transmisión y recepción de los códigos, obteniéndose a partir del producto del tiempo de vuelo de la señal y la velocidad de la luz. El tiempo de vuelo de la señal se mide por comparación del ruido pseudoaleatorio (PRN) generados por el satélite y el receptor. Los códigos generados en el receptor se obtienen a partir del reloj interno que lleva, de la misma manera que el código del satélite se genera a partir del reloj que monta a bordo. Es evidente que ambos relojes tendrán errores (mayores en el receptor, debido a que el oscilador que emplea se suele basar en la oscilación de un cristal de cuarzo, mientras que el satélite emplea un patrón atómico), errores que deben sumarse al hecho de que la trayectoria no es rectilínea, ni la velocidad de la luz es constante durante el tiempo de vuelo, ya que atraviesa diferentes capas de la atmósfera. Para la medida de la pseudodistancia se pueden emplear tanto el código C/A como el P, obteniéndose mayores precisiones al usar el código P. La pseudodistancia es la base para la resolución del problema de navegación.

2.2.1.1 MODELO MATEMÁTICO.

Como se ha mencionado anteriormente, se mide el tiempo de vuelo de la señal generada por el satélite, y se compara con la generada por el receptor en la misma época. Según esto, consideremos un satélite k y un receptor i. En un momento cualquiera se genera la señal de manera simultánea, al llegar al receptor se compara con la generada por éste último, obteniéndose una diferencia de tiempo δt . Debido a que los relojes de satélite y receptor no están en sincronía, la pseudodistacia no se corresponde con la distancia geométrica. Llamemos t^k a la lectura del oscilador del satélite k y t_i a la correspondiente al receptor. Ambos relojes presentan atrasos o adelantos respecto de la escala de

tiempo GPS, siendo $dt^k(t)$ el desfase correspondiente al satélite k y $dt_i(t)$ el del receptor, de manera que (ver figura 2.1):

$$t^k(t) = t^k(GPS) + dt^k(t) \quad [2.1]$$

$$t_i(t) = t_i(GPS) + dt_i(t)$$
 [2.2]

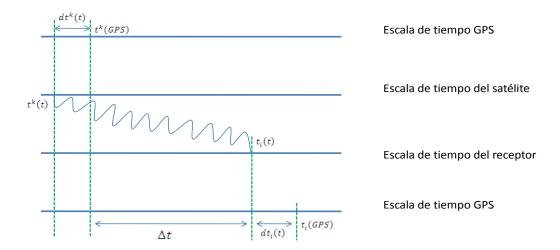


Fig. 2.1: Pseudodistancia

Junto con los retrasos debidos a efectos atmosféricos que hace que la velocidad de la luz no sea constante, los producidos por cuerpos reflectantes en las proximidades de la estación y los debidos a la electrónica del receptor, si tenemos en cuenta las fórmulas [2.1] y [2.2], para obtener el desplazamiento de tiempo Δt que hace coincidir ambas señales, basta con multiplicar por la velocidad de la luz, obteniéndose:

$$c\Delta t = c\Delta t (GPS) + c[dt^{k}(t) - dt_{i}(t)] + cdt_{atmos}(t) + cdt_{retardos}(t)$$
 [2.3]

Los efectos atmosféricos se dividen en dos, a los que denominaremos como I (efecto ionosférico) y T (efecto troposférico). Se hace esta distinción debido a que el comportamiento de la señal es muy diferente en estas dos capas, siendo despreciable el efecto producido por las demás capas de la atmósfera. Si además llamamos p a la distancia geométrica entre satélite y

receptor i.e. la distancia entre el satélite y el receptor en las épocas $t^k(GPS)$ y $t_i(GPS)$, y denotamos como ε a la corrección por ruido blanco (ruido de carácter aleatorio), la ecuación [2.3] queda de la siguiente forma:

$$p_i^k = c\Delta t = \rho_i^i(t) + c[dt^k(t) - dt_i(t)] + I_i^k(t) + T_i^k(t) + \varepsilon_i^k$$
 [2.4]

El objeto del problema geodésico es la resolución de las coordenadas de la estación, que vienen implícitas dentro de la distancia geométrica ρ, que se puede expresar como:

$$\rho_i^k(t) = \sqrt{[x^k(t) - x_i(t)]^2 + [y^k(t) - y_i(t)]^2 + [z^k(t) - z(t)]^2}$$
 [2.5]

donde $[(x^k, y^k, z^k), (x_i, y_i, z_i)](t)$ son las coordenadas cartesianas geocéntricas de satélite y receptor, para una época t, respectivamente. Al sustituir [2.5] en la ecuación [4] obtenemos la expresión completa de la pseudodistancia. Con lo visto anteriormente, se observa que el número de incógnitas de la ecuación es 4, y son las tres coordenadas cartesianas del receptor junto con el error del reloj de éste último. El error del reloj del satélite no es una incógnita, ya que es calculado por el segmento de control y difundido en el mensaje de navegación. Tampoco son incógnitas las efemérides del satélite, que son emitidas en el mensaje de navegación, o bien adquiridas por organismos oficiales, a partir de seguimientos precisos, conociéndose como efemérides precisas, de gran importancia en geodesia.

Con el fin de obtener un menor error entre la escala de tiempo del satélite y la escala de tiempo GPS, se emplea una aproximación polinómica, cuyos coeficientes se calculan y actualizan mediante el seguimiento preciso del satélite desde las estaciones secundarias del sistema. Para dicha aproximación se emplea un polinomio de segundo orden, cuyos coeficientes para el instante t_0 y el instante de observación t:

$$dt^{k}(t) = a_0 + a_1(t - t_0) + a_2(t - t_0)^2$$
 [2.6]

2.2.2 FASE DE LA PORTADORA.

La medida de la fase de la portadora se obtiene por comparación de la fase de la señal portadora recibida en el receptor y la generada por el oscilador del propio receptor. Como ocurría en el caso de la pseudodistancia, se comparan para una época t, si bien cada una de las fases es función del tiempo nominal del oscilador que la genera. En este caso el observable a registrar es la fase de la onda portadora obtenida al reconstruir la portadora recibida, libre de los códigos y mensaje de navegación. Como consecuencia de la mayor frecuencia en comparación con los códigos emitidos, la portadora permite mayores precisiones en la medida de la distancia.

El sistema cuenta el número entero de ciclos que transcurren entre la transmisión y la recepción de la portadora. La distancia se obtiene multiplicando la fase de la señal a la que hay que añadir el número entero de ciclos entre receptor y satélite (a dicho número entero se le conoce como ambigüedad). El principal problema radica en que el número de ciclos que el receptor recibe hasta que adquiere la señal es desconocido. A este número de ciclos se le conoce como ambigüedad, o el valor entero que añadido a la observable de fase registrada por el receptor permite determinar la pseudodistancia. Con lo dicho anteriormente la medida de fase en un cierto instante se compone de un número entero de ciclos y una parte fraccional de ciclo. No se debe confundir ambigüedad con la pérdida de ciclos producida por la interrupción en el seguimiento de la señal, una vez ha sido adquirida.

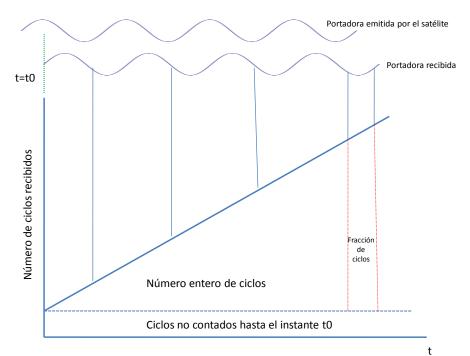


Fig. 2.2 Fase de la portadora.

2.2.2.1 MODELO MATEMÁTICO.

El oscilador del satélite genera una señal sinusoidal cuya frecuencia es la derivada de la fase. Como ocurre con la medida de la pseudodistancia empleando código, los osciladores del receptor y del satélite no están en perfecta sincronía respecto del sistema de tiempo GPS. Así pues, manteniendo la nomenclatura empleada, para un satélite k y un receptor i tendremos que:

$$t^{k}(t) = t^{k}(GPS) + dt^{k}(t)$$
 [2.7]

$$t_i(t) = t_i(GPS) + dt_i(t) \quad [2.8]$$

en donde dt representa los errores del reloj del satélite y receptor, $t^k(t)$ y $t_i(t)$ representan instantes de tiempo registrados en satélite y receptor en sus respectivas escalas, $t^k(GPS)$ y $t_i(GPS)$ los instantes registrados en la escala de tiempo GPS.

Como se ha mencionado anteriormente, existe una componente indeterminada en la medida de la fase conocida como ambigüedad (y que representaremos por N), por lo tanto, la fase de la portadora es:

$$\varphi_{total} = \varphi_{medida} + N(t_0)$$
 [2.9]

siendo la fase medida la diferencia entre la fase que se genera en el receptor i en el instante de recepción de la señal:

$$\varphi_i(t) = f_i t_i(t) + \varphi_i(t_0) = f_i(t + dt_i(t)) + \varphi_i(t_0)$$
 [2.10]

Por otra parte, en la señal generada en el satélite ha de tenerse en cuenta que tarda la onda en recorrer la distancia satélite-receptor t_{ρ} . De manera que la expresión de la fase que produce el oscilador del satélite es:

$$\varphi^{k}(t - t_{\rho}) = f^{k}[t^{k}(t) - t_{\rho}] + \varphi^{k}(t_{0}) = f^{k}[t + dt^{k}(t)] - f^{k}t_{\rho} + \varphi^{k}(t_{0}) \quad [2.11]$$

en donde por comodidad los tiempos tanto del satélite como del receptor en la escala GPS se expresan como t. Teniendo en cuenta las ecuaciones 10 y 11, la fase de la portadora medida es:

$$\varphi_{medida} = \varphi_i^k(t) = \varphi_i(t) - \varphi^k(t - t_\rho) = f_i(t + dt_i(t)) + \varphi_i(t_0) - f^k(t + dt^k(t)) + f^k t_\rho - \varphi^k(t_0)$$
[2.12]

Debido a que los relojes que montan los satélites son patrones atómicos, y estos tienen un gran estabilidad (del orden de $10^{-12}\,s$), la ecuación 2.12 puede simplificarse asumiendo que la pérdida de ciclos (10^{-4}) está por debajo del nivel de ruido que puede discernir el sistema, por lo que se puede asumir la aproximación de considerar que las frecuencias de portadora de satélite y receptor son las mismas, e iguales a la frecuencia nominal f_0 , por lo que la ecuación 2.12 queda de la siguiente forma:

$$\varphi_i^k(t) = f_0[dt_i(t) - dt^k(t)] + f_0t_0 + \varphi_i(t_0) - \varphi^k(t_0) + \epsilon_i^k$$
 [2.13]

Si a la ecuación 2.13, en la que se han incluido los errores, le sumamos la ambigüedad se obtiene:

$$\varphi_i^k(t) = f_0[dt_i(t) - dt^k(t)] + f_0t_0 + \varphi_i(t_0) - \varphi^k(t_0) + \epsilon_i^k + N_i^k(t_0)$$
 [2.14]

La ecuación 2.14 representa el número de ciclos, por lo tanto, si se quiere obtener la fase total es necesario multiplicar por la longitud de onda:

$$\Phi(t)_{i}^{k} = ct_{\rho} + \lambda [\varphi_{i}(t_{0}) - \varphi^{k}(t_{0})] + c[dt_{i}(t) - dt^{k}(t)] + \lambda N_{i}^{k}(t_{0}) + \epsilon_{i}^{k}$$
 [2.15]

en donde ct_{ρ} representa la distancia geométrica que separa satélite y receptor, y $\lambda[\varphi_i(t_0)-\varphi^k(t_0)]$ es la fase inicial y ϵ_i^k es una corrección en ruido de carácter aleatorio, diferente al de la ecuación 2.13, ya que ahora se expresa en unidades de distancia. Como en esta ecuación interviene también la velocidad de la luz, que no es constante durante el proceso de posicionamiento ya que atraviesa distintas capas de la atmósfera, la ecuación 2.15 se ve afectada por efectos de propagación atmosférica. Aplicando lo dicho, 2.15 queda de la siguiente manera:

$$\Phi(t)_{i}^{k} = \rho_{i}^{k}(t) + \lambda [\varphi_{i}(t_{0}) - \varphi^{k}(t_{0})] + c[dt_{i}(t) - dt^{k}(t)] + \lambda N_{i}^{k}(t_{0}) - I_{i}^{k}(t) + T_{i}^{k}(t) + \epsilon_{i}^{k}$$
 [2.16]

el signo negativo en el retardo ionosférico se debe a la dispersividad de la ionosfera. En la ecuación 2.16 las incógnitas son el estado del reloj o error del reloj del receptor, la ambigüedad inicial (constante para cada satélite mientras no se pierda la señal) y las coordenadas del receptor, intrínsecas en la distancia geométrica satélite receptor.

2.3 Errores.

En el posicionamiento por satélite intervienen múltiples elementos, como son los propios satélites, el retardo que sufre la señal al atravesar las diferentes capas de la atmósfera, la estación receptora y su entorno etc. Todos estos factores añaden errores y limitan la precisión de la solución de posicionamiento en mayor o menor medida. Incluso los relojes de los satélites, que a pesar de su gran estabilidad, pueden incurrir en discrepancias que se convierten en errores.

Algunos de estos errores pueden ser modelados y ser compensados, incluso, mediante combinaciones de observables. Parte de estos errores pueden ser eliminados. Efectos derivados de la geometría de la constelación en el instante de la toma datos son otra fuente de imprecisiones, ya que suele haber disponibles más satélites de los necesarios para posicionamiento, empleando algunos e ignorando otros. Si los satélites empleados están muy próximos entre sí, las esferas de intersección se cruzarán formando ángulos con poca diferencia, incrementando el error (lo ideal es que se intersecaran formando ángulos rectos).

Los problemas no terminan cuando la señal llega al receptor. El software del receptor, el recorrido de la señal desde el centro de fase de la antena hasta la unidad de correlación entre otros, supones un retardo, que a pesar de ser muy pequeño, ha de ser tenido en cuenta en posicionamiento de altas precisiones. Los errores pueden ser agrupados en tres clases:

- Errores debidos a las efemérides y estados del reloj de los satélites.

- Errores debidos a la propagación de la señal a través de la atmósfera.
- Errores debidos al receptor y su entorno.

2.3.1 ERRORES DEBIDOS A LAS EFEMÉRIDES Y ESTADOS DEL RELOJ DE LOS SATÉLITES.

Como se ha mencionado anteriormente, son necesarios cuatro satélites y sus distancias al receptor para obtener la solución de posicionamiento, o dicho de otra manera, es necesario conocer las coordenadas cartesianas de los cuatro satélites y el tiempo de vuelo de la señal emitida por cada uno de ellos. Además, las posiciones relativas entre los satélites deben tener una configuración determina, en aras de minimizar los errores. Los diferentes errores asociados a las efemérides son:

- Geometría de la constelación. El vector que determina el receptor con cada uno de los satélites forma un poliedro cuya forma varía de manera continua, pero muy lentamente debido a la baja velocidad relativa de los satélites. Se expresa mediante un parámetro llamado DOP (Dilution of Precision). La Dilución de la Precisión es una medida de la fortaleza de la geometría de los satélites, y está relacionada con la distancia entre estos y su posición en el espacio, es decir, está considerando la influencia que en el error final de posición está teniendo la configuración espacial de los satélites y el usuario. Dentro de este apartado se incluye también el efector Wind Up (Kouba et al. 2009) relativo a la actitud del satélite en el momento de transmisión de la señal.
- Efemérides de los satélites. En el problema del posicionamiento es necesario conocer las coordenadas cartesianas de cada satélite empleado en la solución, o su vector geocéntrico. Dicho vector con sus variaciones conforman las efemérides del satélite. Estas efemérides pueden ser transmitidas o precisas. Las primeras son enviadas por cada satélite en el mensaje de navegación, e inducen a una mayor incertidumbre en la posición que las precisas. Las precisas se calculan a posteriori por varios organismos a partir de los datos registrados por multitud de estaciones repartidas por todo el mundo (Hofmann-Wellenhof et al. 1992) no estando disponibles hasta varios días después de la observación, por lo que no pueden ser empleadas en navegación.
- Osciladores de los satélites. A pesar de estar equipados con relojes atómicos, éstos presentan una deriva que no se puede modelar, desplazándose con respecto a la escala de tiempo GPS que

producen errores que se propagarán en la solución de posicionamiento. La corrección que se aplica a los relojes de los satélites es calculada por el segmento de control, e inyectada en los satélites junto con las efemérides por la Master Control Station.

2.3.2 ERRORES DEBIDOS A LA PROPAGACIÓN DE LA SEÑAL A TRAVÉS DE LA ATMÓSFERA.

Como consecuencia que la atmósfera no es homogénea, y se compone de distintas capas, la señal sufre un retardo debido a las refracciones que sufre en su trayectoria y que ha de ser tenido en cuenta para la solución de posicionamiento. Las dos capas que afectan a la propagación de la señal son la troposfera y la ionosfera.

- Efecto troposférico. La troposfera es la capa de la atmósfera comprendida hasta los 11 km de altura, estando dividida la atmósfera en dos regiones (Saatamoinen, 1972). Existen modelos para tratar el retraso que sufre la señal al atravesar la parte seca de la troposfera, siendo más difícil la modelización de la parte húmeda debido a las variaciones del vapor de agua en el tiempo y el espacio. Para la estimación del retardo troposférico, el IGS proporciona para cada estación información del retardo troposférico cenital, o ZPD (Zenit Path Delay).
- Efecto ionosférico: La ionosfera es la capa de la atmósfera que se extiende desde los 50 km hasta los 1000 km. En ella, el aire está ionizado debido a la acción de la radiación solar. Esta característica la convierte en una capa dispersiva, afectando de manera diferente a la parte portadora y modulada de la señal GPS, de hecho, los efectos provocados en la señal GPS son opuestos, sufriendo los códigos un retraso en su velocidad de propagación, aumentando por el contrario la de la portadora. El efecto ionosférico será mayor durante el día, ya que la acción de la radiación solar es mayor, siendo el factor que determina este efecto el conocido como TEC, o contenido total de electrones. Αl emplear receptores bifrecuencia reduce considerablemente el efecto ionosférico, gracias a que emplea combinaciones lineales de las frecuencias L1 y L2.

2.3.3 ERRORES DEBIDOS AL RECEPTOR Y SU ENTORNO

El emplazamiento donde se sitúa un receptor no es fijo, debido a que están afectados por los movimientos de deformación que sufre la Tierra. A esto se le suma el hecho de que la Tierra tiene una cierta elasticidad, que se traduce en una deformación conocida como marea terrestre.

La carga de marea oceánica produce variaciones en las posiciones de puntos cercanos a la costa, reflejándose esta influencia principalmente en la coordenada vertical. La causa de estas mareas está en la atracción que ejerce la luna sobre las masas terrestres, más evidentes sobre los océanos, y despreciable sobre los continentes.

En menor medida también hay atracción del Sol y de los planetas mayores, en especial Júpiter, debido a que existen perturbaciones debidas principalmente al efecto de la Luna, y en una proporción mucho menor al resto de planetas del sistema solar. Así mismo, las grandes masas continentales sufren una deriva, que aunque pequeña en términos anuales (unos pocos milímetros), afectan al posicionamiento con fines geodésicos.

Existen ocasiones en las que la señal que se recibe en la antena del receptor no llega de manera directa desde el satélite, debido a que rebota en elementos del entorno del receptor como pueden ser árboles o edificios. La consecuencia de la recepción de estas señales es que la pseudodistancia no es la real que separa a la dupla satélite-receptor, es una distancia mayor.

A esto se conoce como efecto multipath. Además, disfunciones en el software y osciladores del receptor pueden provocar que se pierda el seguimiento continuo de la señal. A este efecto se le conoce como pérdida de ciclo.

Además de los errores enunciados anteriormente, existe un tipo de error asociado al hecho de que el sistema de referencia terrestre tiene como origen en centro de la Tierra y se considera fijo, mientras que cada satélite materializa un sistema de referencia acelerado. Por ello ha de tenerse en cuenta la Teoría Especial de la Relatividad, cuyos efectos influyen en la órbita de cada satélite, la propagación de la señal y en los osciladores de satélites y receptores. El efecto principal se produce sobre la frecuencia de dichos osciladores, compensándose al sintonizarlo a una frecuencia ligeramente diferente antes del lanzamiento. Estos efectos son muy pequeños.

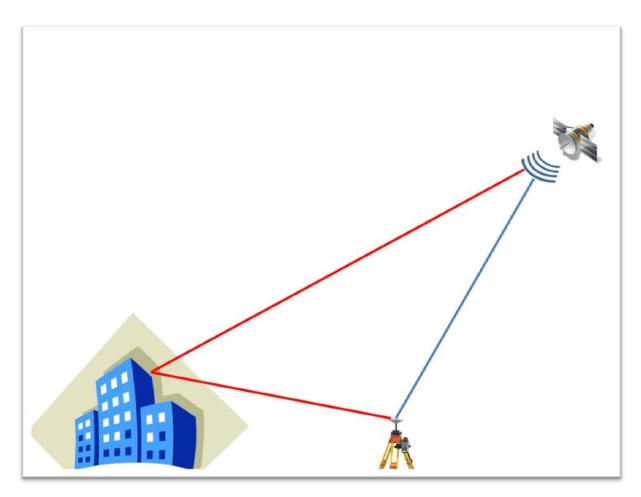


Fig. 2.3: Efecto Multipath

Capítulo 3. Geodesia y Aplicaciones

En las aplicaciones cotidianas (como la navegación por ejemplo), el posicionamiento absoluto por satélite se consigue con los modelos de fase y pseudodistancia descritos en el capítulo 2. Sin embargo, para fines geodésicos, se requieren precisiones mucho mayores, no realizables con la metodología descrita hasta el momento. Estas precisiones sólo se logran obtener empleando posicionamiento relativo entre las distintas estaciones de una red geodésica, para a partir de unas estaciones fijas dotar a todas las estaciones de la red de coordenadas absolutas. De lo que se trata es de determinar los incrementos entre las coordenadas cartesianas geocéntricas entre las estaciones que componen la red. Esto se logra gracias a que la constelación GPS está constituida de manera que un satélite sea observado por varias estaciones al mismo tiempo, lo que permite que una determinada observable sea registrada por varios receptores de manera simultánea. La importante mejora en la precisión se debe a que se eliminan o reducen considerablemente algunas de las fuentes de error (como el efecto ionosférico para estaciones lo suficientemente próximas, e incluso el troposférico).

3.1 Simples diferencias.

La ecuación de simples diferencias es el resultado de la observación de un satélite k desde dos estaciones i, j cuya posición relativa se quiere determinar. Recordando la ecuación 2.16 del capítulo 2 se tiene que, la medida de fase es:

$$\begin{split} \Phi(t)_i^k &= \rho_i^k(t) + \lambda [\varphi_i(t_0) - \varphi^k(t_0)] + c[dt_i(t) - dt^k(t)] + \lambda N_i^k(t_0) - I_{i,j}^k(t) \\ &+ T_i^k(t) + \epsilon_i^k \end{split}$$

si se halla la diferencia entre las medidas de fase de ambas estaciones se obtiene que:

$$\Phi(t)_{j}^{k} - \Phi(t)_{i}^{k} = \rho_{j}^{k}(t) - \rho_{i}^{k}(t) + \lambda \left[\varphi_{j}(t_{0}) - \varphi_{i}(t_{0})\right] + c\left[dt_{j}(t) - dt_{i}(t)\right] + \lambda \left[N_{j}^{k}(t_{0}) - N_{i}^{k}(t_{0})\right] - \left[I_{j}^{k}(t) - I_{i}^{k}(t)\right] + \left[T_{j}^{k}(t) - T_{j}^{k}(t)\right] + \left[\epsilon_{j}^{k} - \epsilon_{i}^{k}\right]$$
[3.1]

empleando la siguiente notación:

$$U_{i,j}^k = U_j^k - U_i^k$$
 [3.2]

la ecuación 3.1 queda abreviada de la siguiente forma:

$$\Phi(t)_{i,j}^{k} = \rho_{i,j}^{k}(t) + \lambda \varphi_{i,j}(t_0) + cdt_{i,j}(t) + \lambda N_{i,j}^{k}(t_0) - I_{i,j}^{k}(t) + T_{i,j}^{k}(t) + \epsilon_{i,j}^{k}$$
 [3.3]

donde las incógnitas son las posiciones de cada estación, la diferencia de los estados de los relojes de los receptores y la diferencia de ambigüedades. Como se aprecia, presenta la ventaja de que elimina el error debido al estado del reloj del satélite respecto a la escala de tiempo GPS. Además, si las estaciones están lo suficientemente próximas, los retardos atmosféricos también se eliminan.

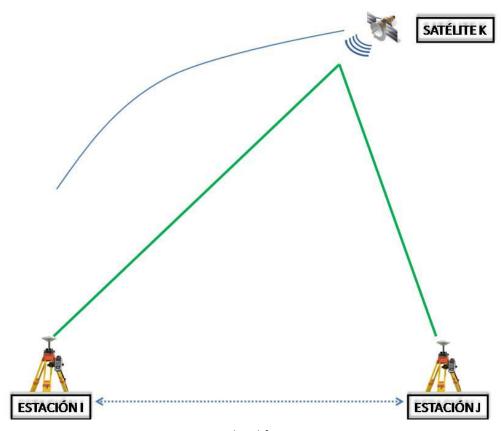


Fig. 3.1: Simples diferencias.

3.2 Dobles diferencias.

El algoritmo de dobles diferencias se obtiene al hallar la diferencia entre dos ecuaciones de simples diferencias simultáneas de dos estaciones i, j a partir de observaciones realizadas a dos satélites k, l. en el apartado anterior se ha llegado a la ecuación de diferencia simple entre dos estaciones a partir de la medición de las observables a un satélite (ecuación 3.3). Teniendo en cuenta dicha ecuación se obtendrá:

$$\begin{split} \Phi(t)_{i,j}^{k} - \Phi(t)_{i,j}^{l} \\ &= \rho_{i,j}^{k}(t) - \rho_{i,j}^{l}(t) + \lambda [\varphi_{i,j}^{k}(t_{0}) - \varphi_{i,j}^{l}(t_{0})] + \lambda [N_{i,j}^{k}(t_{0}) - N_{i,j}^{l}(t_{0})] \\ &- [I_{i,j}^{k}(t) - I_{i,j}^{l}(t)] + T_{i,j}^{k}(t) - T_{i,j}^{l}(t) + \epsilon_{i,j}^{k} - \epsilon_{i,j}^{l} \quad [3.4] \end{split}$$

ecuación la 3.3, que si se emplea la notación:

$$U_{i,j}^{l,k} = U_{i,j}^k - U_{i,j}^l = U_i^k - U_i^k - \left[U_i^l - U_i^l\right]$$
 [3.5]

se obtiene:

$$\Phi_{i,j}^{l,k}(t) = \rho_{i,j}^{l,k}(t) + \lambda [\varphi_{i,j}^{l,k}(t_0)] + \lambda [N_{i,j}^{l,k}(t_0)] - I_{i,j}^{l,k}(t) + T_{i,j}^{l,k}(t) + \epsilon_{i,j}^{l,k}$$
 [3.6]

Como se aprecia en la ecuación 3.6, la ventaja del empleo de dobles diferencias es que desaparece de la ecuación el estado del reloj de los receptores de las dos estaciones. Además, como ocurre en el caso de las simples diferencias, si las estaciones receptoras están próximas, se puede asumir la aproximación de que los errores debidos al retardo atmosférico también se eliminan. En este caso las incógnitas siguen siendo las distancias estación-satélite, y la ambigüedad inicial de ambas estaciones.

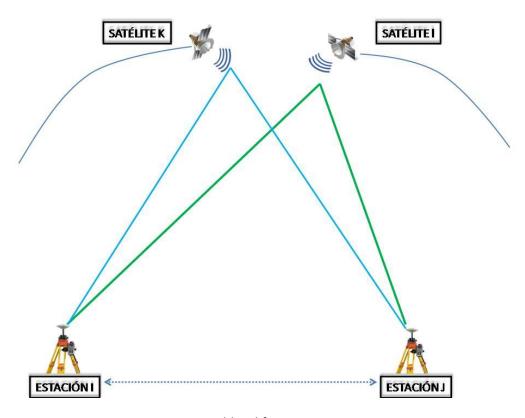


Fig. 3.2: Dobles diferencias.

3.3 Triples diferencias.

En este caso, se trata de establecer diferencias entre las ecuaciones de dobles diferencias, para dos épocas distintas t1 y t2. Partiendo de la ecuación 3.6 se tiene que:

$$\begin{split} \Phi_{i,j}^{l,k}(t2) - \Phi_{i,j}^{l,k}(t1) \\ &= \rho_{i,j}^{l,k}(t2) - \rho_{i,j}^{l,k}(t1) - \left[I_{i,j}^{l,k}(t2) - I_{i,j}^{l,k}(t1) + \left[T_{i,j}^{l,k}(t2) - T_{i,j}^{l,k}(t1)\right] \\ &+ \epsilon_{i,j}^{l,k} \quad [3.7] \end{split}$$

Si por comodidad se emplea la notación:

$$U_{i,j}^{l,k}(t12) = U_{i,j}^{l,k}(t2) - U_{i,j}^{l,k}(t1)$$
 [3.8]

y se tiene en cuenta la notación empleada en 3.5, la ecuación 3.7 queda de la siguiente forma:

$$\Phi_{i,j}^{l,k}(t12) = \rho_{i,j}^{l,k}(t12) - I_{i,j}^{l,k}(t12) + T_{i,j}^{l,k}(t12) + \epsilon_{i,j}^{l,k}$$
 [3.8]

El empleo del algoritmo de triples diferencias presenta la gran ventaja de que se eliminan las ambigüedades iniciales. Además, no está afectado por las posibles interrupciones que se puedan producir en la recepción de la señal. Sin embargo, a pesar de que las triples diferencias eliminan sistematismos, plantea el inconveniente de que el número de observaciones a realizar se incrementa, aumentando el ruido, con la consiguiente disminución de la precisión en el posicionamiento. Suele ser empleado como un proceso de cálculo previo, pensado para obtener unos valores aproximados de las ambigüedades iniciales de fase, para lograr el cálculo definitivo de dichas ambigüedades mediante un proceso de dobles diferencias.

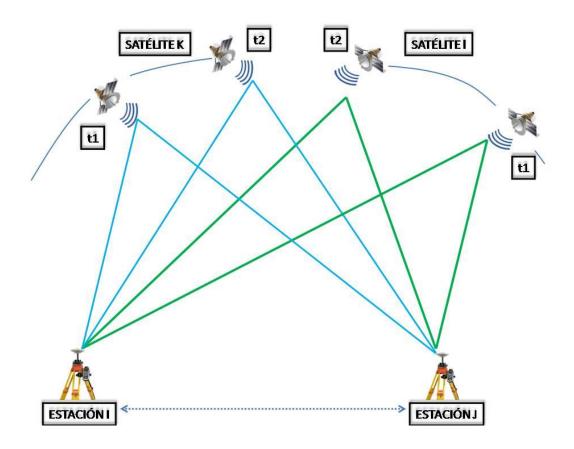


Fig. 3.3: Triples diferencias.

3.4 Órbitas precisas.

Como se ha visto, el fundamento del posicionamiento por satélite parte del conocimiento preciso de la posición de los satélites de la constelación. La precisión de las efemérides radiodifundidas en el mensaje de navegación es suficiente para aplicaciones en tiempo real, donde no se requieren grandes precisiones. En aplicaciones geodésicas, donde se necesitan grandes niveles de precisión, es necesario emplear efemérides precisas. Para ello, existen distintas instituciones que se dedican al cálculo de las órbitas de los satélites con gran precisión, además de otros productos asociados, como son el estado del reloj de los satélites, las coordenadas geocéntricas de las estaciones de seguimiento, los parámetros de orientación de la Tierra y los parámetros atmosféricos. Entre las distintas instituciones se encuentran el IGS (International GNSS Service) y el JPL (Jet Propulsion Laboratory). La información precisa que estas instituciones producen es empleada para mejorar la precisión en el posicionamiento. Ahora bien, debido al proceso de cálculo necesario para la elaboración de estos productos, los conocidos como GPS Precise Products no están disponibles hasta pasado un cierto tiempo. Según esto se pueden diferenciar tres tipos de productos, a saber: ultrarápidas, rápidas y finales. Existen diferencias entre las producidas por el IGS y el JPL. Las características de cada una de las anteriores son:

- Ultra-rápidas: El IGS publica las efemérides ultra-rápidas cuatro veces al día (a las 0300, 0900, 1500 y 2100 UT) y contienen 48 horas de observaciones. La primera mitad son calculadas a partir de observaciones y la segunda mitad son predicciones. Los archivos se nombran de acuerdo a la hora correspondiente a la mitad de tiempo entre publicación (00, 06, 12 y 18 UT). La primera mitad calculada ofrece una precisión de 3 cm en la órbita y 150 ps (pico-segundos) en el estado del reloj, mientras que la segunda mitad (predicha) ofrece una precisión de 5 cm en la órbita y 3 ns (nano-segundos) en el estado del reloj (igscb.jpl.nava.gov, 2014). Por su parte el JPL trabaja con una ventana de 30 horas, en la que los datos son actualizados cada hora. El prefijo del nombre de cada producto (YYYY-MM-DD) es consistente con la ventana en la que se encuentra. La excepción viene en el periodo entre las 0000 y las 0300 horas, donde los productos de ese día y el anterior son actualizados. Ofrecen una precisión de 3 cm.
- Rápidas: Las efemérides rápidas del IGS se publican cada 17 horas aproximadamente, ofreciendo una precisión en la órbita de 2,5 cm y 75 ps en el estado del reloj del satélite. El JPL las publica a las 1600 UTC del día siguiente con una precisión de 3,5 cm.
- Finales: Las efemérides finales del IGS se publican cada 7 días, aunque se trata de las efemérides correspondientes a unos 12 días antes. aproximadamente, con unas precisiones de 2,5 cm en la órbita y 75 ps en el estado del reloj, mientras que el JPL las publica cada 14 días con una precisión en la órbita de 2,5 cm.

Efemérides GPS	Precisión	Periodo de latencia	
Ultra-rápida (predicha)	5 cm/3 ns	Tiempo real	
Ultra-rápida (observada)	3 cm/150 ps	3-9 horas	
Rápida	2,5 cm/75 ps \mp 25 σ	17-41 horas	
Final	2,5 cm/75 ps \mp 20 σ	12 días	

Tabla. 3.1: Efemérides precisas IGS.

Efemérides GPS	Precisión	Periodo de latencia
Ultra-rápida	5 cm	< 2 horas
Rápida	3,5 cm	Día siguiente 1600 UTC
Final	2,5 cm	< 14 días

Tabla. 3.2: Efemérides precisas JPL.

Capítulo 4. Evolución PPP.

Como se ha visto, debido a las numerosas fuentes de error que afectan a las medidas GNSS, se usan una serie de métodos diferenciales, gracias al empleo de receptores con coordenadas conocidas, con el objetivo de mitigar dichos errores. Dependiendo de varios factores, se pueden alcanzar hasta precisiones sub-centimétricas, estimando las coordenadas con el postprocesado en conjunto de las observaciones de varios receptores, o bien empleando datos de receptores de referencia transmitidos, obteniendo coordenadas relativas en tiempo real. Sin embargo, no siempre es posible recurrir a estos métodos, como en el caso de receptores ubicados en puntos aislados. Además, el uso de las dobles y triples diferencias aumenta mucho la carga de procesado que soportan los ordenadores, por lo que también se busca una simplificación de los cálculos, así como una optimización de los tiempos de uso de los procesadores. Una forma de obtener esto es recurrir al uso de un único receptor, empleando en lugar de los datos del mensaje de navegación, productos precisos como los mencionados en el capítulo anterior. Además, al emplear receptores bi-frecuencia en conjunción con modelos estocásticos y funcionales adecuados, permiten alcanzar precisiones centimétricas. Si bien en un principio, la estrategia PPP se concibió como un método de posicionamiento absoluto estático, posteriores desarrollos han permitido el empleo de este método en aplicaciones cinemáticas. Al emplear esta técnica, no es posible hacer ningún tipo de simplificaciones, teniendo que ser empleadas todas las correcciones conocidas a las observaciones, y dichas observaciones han de ser consistentes. Al no usar dobles diferencias en los algoritmos PPP, no se puede asumir que el retardo ionosférico se cancela, lo que lleva al empleo de receptores bifrecuencia (en geodesia siempre se emplean receptores de este tipo) que permiten crear combinaciones lineales de las observaciones de código y fase, permitiendo eliminar los efectos ionosféricos. Las ambigüedades enteras en L1 y L2 también son combinadas en una constante racional, lo que contribuye a absorber retardos creados por hardware en satélites y receptores que pueden cambiar con el tiempo.

4.1 PPP.

La técnica PPP basa su funcionamiento en una red mundial de receptores, que se emplea para estimar los parámetros de los receptores y transmisores a lo largo de un periodo de tiempo, que normalmente se toma como Δ =24 horas. Entre los parámetros a estimar de los receptores, se encuentran entre otros las coordenadas cartesianas, el ZPD (Zenith Path Delay) o el estado del reloj entre otros, mientras que los parámetros a estimar que afectan a los transmisores se encuentran los parámetros de radiación

solar, el estado del reloj del satélite etc... Debido a que las variaciones de los estados de los relojes de receptores y satélites se asemejan al ruido blanco en escalas de tiempo de minutos, para el periodo seleccionado Δ , es necesario recurrir a variaciones estocásticas de dichos relojes.

Supongamos que la red global se compone de R receptores, empleados para estimar los parámetros anteriormente mencionados, P son las estimaciones de los parámetros del transmisor y Q los parámetros transmitidos del transmisor. Al tomar la aproximación P=Q, al incluir un nuevo receptor en la red (el receptor cuya posición quiero hallar) sus datos no tendrán efectos sobre los parámetros estimados de los transmisores, de manera que la forma más efectiva para estimar los parámetros de este nuevo receptor, será fijar los parámetros de los transmisores como P, que ya han sido previamente estimados por la red global. Al fijar las órbitas y los estados de los relojes de los satélites respecto a un marco de referencia fijo con la Tierra a unos parámetros determinados, se evita el realizar el cálculo de todos los receptores simultáneamente, es decir: se estiman los parámetros específicos de un receptor en concreto mediante el análisis de la pseudodistancia de ese receptor, fijando los parámetros del transmisor a los valores radiodifundidos en el mensaje de navegación. Como es obvio, los resultados obtenidos presentan limitaciones, debido al ruido inherente de la pseudodistancia (de unos 40 cm), y en especial a la precisión de las órbitas de los satélites difundidas en el mensaje de navegación (de unos pocos metros) y al estado del reloj del satélite (decenas de metros). Para solventar esto último, se emplean medidas de fase y pseudodistancia, y mejorando la calidad de los parámetros.

4.2 Método.

Como se ha mencionado anteriormente, la se supone una red global compuesta por R receptores. La situación idónea sería una distribución uniforme de los receptores a lo largo del planeta. Desafortunadamente esto no es así, ya que existen zonas de la Tierra con una alta densidad de receptores, mientras que por el contrario, existen amplias extensiones sin apenas equipos.

La red se divides en dos, una primera parte formada por un número S de receptores, que se emplea para estimar los parámetros de los satélites, orientación de la Tierra y los parámetros de esos S receptores. Una vez hecho esto, se analizan por separado los datos de los restantes R-S receptores. Al procesar uno a uno los receptores se facilita la detección de problemas relativos a un receptor en concreto. Para fines geodésicos, los valores rms de las diferencias entre los valores modelizados y los medidos debe encontrarse entre 5 y 10 mm en el caso de la fase, y entre 50 y 100 cm para la pseudodistancia. Cuando uno de los receptores muestra valores superiores, es indicativo de que ocurre un problema, que debe ser aislado y estudiado para

determinar su causa (efecto multipath, fallos de hardware, interferencia por radiofrecuencia etc...).

4.2.1 MARCO DE REFERENCIA.

En el procedimiento de cálculo, se da por supuesto que las coordenadas de ciertos emplazamientos se conocen con la suficiente precisión (errores inferiores al centímetro), de manera que se pueden tomar como fijas a la hora de realizar el análisis de una red global. Está metodología presenta un serio inconveniente, ya que se asumen errores en las coordenadas fiduciales de las estaciones que se toman como fijas, distorsionando las órbitas de los satélites de la constelación con la que se trabaje de una forma difíciles de compensar.

Los errores antes mencionados se manifiestan principalmente de dos formas. Debido a que el ITRF (International Terrestrial Reference Frame) no permanece fijo en el tiempo, sino que evoluciona, es común que aparezcan discontinuidades en las situaciones de las posiciones fiduciales con el cambio del marco de referencia. Dichas discontinuidades no están relacionadas con movimientos reales de los emplazamientos, indicando un cambio en las posiciones que se asumieron como fijas. Las discontinuidades suelen ser pequeñas, normalmente menores de un centímetro, de manera que pueden ser corregidas aplicando una transformación de Helmert de 7 parámetros (3 giros, 3 traslaciones y un cambio de escala). La no reducción de las discontinuidades trae consigo efectos negativos en las series temporales obtenidas al analizar observaciones fijando las posiciones de los satélites, ya que esas discontinuidades se asumen, traduciéndose en variaciones de varios milímetros en los resultados.

Existe un segundo tipo de errores asociados a las alturas de las antenas de los emplazamientos fiduciales, que no pueden ser completamente eliminados sin recurrir a costosos métodos de reprocesamiento de los datos de la red global. Con el objeto de evitar las consecuencias de los errores fiduciales, se establecen una serie de incertidumbres en las coordenadas fiduciales (en torno a 1 km.). Al hacer esto, la solución se refiere a un marco de referencia muy pobremente definido (de constreñimientos mínimos), que absorberá los errores derivados de las asunciones hechas en las coordenadas de fiduciales de los emplazamientos, evitando además el reproceso de todas las estaciones de la red global. De esta manera, se puede crear un marco de referencia particular transformando los parámetros estimados de los receptores. Con esto, los parámetros estimados de los emplazamientos, cuyos parámetros del transmisor se encuentran en una red libre, se encuentran referidos al mismo marco que la solución libre original. En este punto, una simple transformación bastará para hacer coincidir este marco de referencia libre con el ITRF deseado. Esta transformación es facilitada por los centros de análisis que producen las órbitas y estados del reloj asociados a la red libre.

4.2.2 ECUACIONES DE OBSERVACIÓN.

Como se ha mencionado en el apartado anterior, se emplea una combinación lineal de las observaciones de pseudodistancias y fases de portadora. Estas observaciones están relacionadas con la posición del usuario, el estado del reloj del receptor, retardo troposférico (componente seca y húmeda de la troposfera), ambigüedad y ruido de la medición, según las siguientes ecuaciones de observación (Kouba y Héroux, 2000):

$$l_P = \rho + C(dt - dT) + T_r + \epsilon_p$$
 [4.1]

$$l_{\phi} = \rho + C(dt - dT) + T_r + N\lambda + \epsilon_{\phi}$$
 [4.2]

donde:

 l_P es la combinación ionosfera-libre de las frecuencias L1 y L2 para medidas de pseudodistancias.

 l_ϕ es la combinación ionosfera-libre de las frecuencias L1 y L2 para medidas de fase de la portadora.

dT es el offset del reloj del receptor de la estación respecto a la escala de tiempo GPS.

dt es el offset del reloj del satélite respecto a la escala de tiempo GPS.

c es la velocidad de la luz.

 T_r es el retraso que sufre la señal debido a la troposfera.

λ es la longitud de onda de la fase de la portadora o su combinación ionosferalibre.

N es la ambigüedad de la combinación ionosfera-libre de la fase de la portadora.

 ϵ son los distintos tipos de ruido que afectan a las medidas, incluyendo el efecto multipath.

 ρ es la distancia geométrica que separa la posición del satélite en la época de transmisión t y la posición de la estación en la época de recepción.

Si se emplean productos precisos como los mencionados del IGS o el JPL el estado del reloj del satélite puede fijarse (el JPL proporciona el estado de los relojes de todos los satélites para cada día), de manera que se elimina al hacer la combinación lineal. Además, el retardo troposférico puede tratarse como el producto de una función de mapeo M por el retraso cenital troposférico (zpd), obteniendo, a partir de las fórmulas 4.1 y 4.2 lo siguiente:

$$f_P = \rho + CdT + Mzpd + \epsilon_p - l_p = 0$$
 [4.3]

$$f_{\phi} = \rho + CdT + Mzpd + N\lambda + \epsilon_{\phi} - l_{\phi} = 0$$
 [4.4]

La resolución de las ecuaciones 4.3 y 4.4 permiten estimar las coordenadas del receptor, el estado de su reloj y en retardo troposférico en el emplazamiento, siendo dichas coordenadas consistentes con el marco de referencia global que implica fijar los productos orbitales precisos. Este método presenta el principal inconveniente de que no es posible fijar las ambigüedades de las frecuencias portadoras.

4.2.3 MODELO DE AJUSTE.

Las ecuaciones 4.3 y 4.4 se pueden linealizar empleando valores aproximados de parámetros incógnita y observaciones (X^0, l) según la siguiente forma matricial (Kouba y Héroux, 2000):

$$A\delta + W - V = 0 ag{4.5}$$

donde A es la matriz de diseño, δ es el vector de correcciones a los parámetros incógnita X, W es el vector de términos libres y V el vector residuales. En el caso del posicionamiento de punto preciso el vector X está compuesto por los siguientes parámetros: coordenadas de la estación, estado del reloj del receptor retardo cenital troposférico y las ambigüedades de fase. Para formar la matriz de diseño se hallan las derivadas parciales de las ecuaciones de observación respecto del vector X, obteniéndose:

$$A = \begin{pmatrix} \frac{\delta f(X, l_{P})}{\delta X_{x}} & \frac{\delta f(X, l_{P})}{\delta X_{y}} & \frac{\delta f(X, l_{P})}{\delta X_{z}} & \frac{\delta f(X, l_{P})}{\delta X_{dt}} & \frac{\delta f(X, l_{P})}{\delta X_{zpd}} & \frac{\delta f(X, l_{P})}{\delta X_{N^{j}}} \\ \frac{\delta f(X, l_{\phi})}{\delta X_{x}} & \frac{\delta f(X, l_{\phi})}{\delta X_{y}} & \frac{\delta f(X, l_{\phi})}{\delta X_{z}} & \frac{\delta f(X, l_{\phi})}{\delta X_{dt}} & \frac{\delta f(X, l_{\phi})}{\delta X_{zpd}} & \frac{\delta f(X, l_{\phi})}{\delta X_{N^{j}}} \end{pmatrix}$$

$$\text{con: } \frac{\delta f}{\delta X_x} = \frac{x - X_S}{\rho}, \quad \frac{\delta f}{\delta X_y} = \frac{y - Y_S}{\rho}, \quad \frac{\delta f}{\delta X_z} = \frac{z - Z_S}{\rho}, \quad \frac{\delta f}{\delta X_{dt}} = C, \quad \frac{\delta f}{\delta X_{zpd}} = M, \quad \frac{\delta f}{\delta N^j} = 0 \text{ \'o } 1$$

У

$$X = \begin{pmatrix} x \\ y \\ z \\ dt \\ zpd \\ N^{j} \end{pmatrix}$$

El siguiente paso es un ajuste por mínimos cuadrados, con restricciones a-priori, donde la matriz de las observaciones ponderadas es una matriz

diagonal igual a $\left(\frac{\sigma_0}{\sigma_P}\right)^2$ y $\left(\frac{\sigma_0}{\sigma_\phi}\right)^2$ y σ_0 es la desviación estándar de los pesos, siendo σ_P y σ_ϕ las desviaciones estándar de las observaciones de la pseudodistancia y fase, respectivamente.

4.2.4 PROCEDIMIENTO DE AJUSTE.

El modelo de ajuste descrito en el apartado anterior puede aplicarse de diversas maneras, ya sea en un solo paso o en un ajuste secuencial que implique el uso de un filtro. El realizar el ajuste en un solo paso presenta el inconveniente de la carga computacional que se soporta. Por su parte, el empleo de filtros lleva a un proceso mucho más eficiente, gracias a que los parámetros sólo aparecen en la época de observación. Por contra, las aplicaciones de filtrado requieren de suavizados para aquellos parámetros que no permanecen constantes con el tiempo, que son todos menos la ambigüedad, siempre que no haya pérdida de ciclos. Los procesos de filtrado pueden modelar estas variaciones temporales mediante el uso de los apropiados procesos estocásticos, que también actualizan las varianzas de los parámetros.

En el caso de que no haya observaciones entre épocas, los parámetros iniciales estimados en una determinada época i serán iguales que los obtenidos en una época anterior i-1 (Kouba, 2003):

$$X_i^0 = \widehat{X_{i-1}} \tag{4.6}$$

Para propagar la información que da la covarianza desde la época i-1 a la época i, durante un intervalo de tiempo Δt , $C_{\widehat{X_{l-1}}}$ debe ser actualizada a fin de incluir el ruido que afecta al proceso y que es representado por la matriz de covarianza $C_{\epsilon_{\Delta t}}$:

$$P_{X_i^0 = [C_{\widehat{X}_{i-1}} + C_{\in_{\Delta t}}]^{-1}}$$
 [4.7]

donde:

$$C_{\in_{\Delta t}} = \begin{pmatrix} C_{\in(x)_{\Delta t}} & 0 & 0 & 0 & 0 & 0 \\ 0 & C_{\in(y)_{\Delta t}} & 0 & 0 & 0 & 0 \\ 0 & 0 & C_{\in(z)_{\Delta t}} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{\in(dt)_{\Delta t}} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{\in(zpd)_{\Delta t}} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{\in(zpd)_{\Delta t}} \end{pmatrix}$$
[4.8]

Como las ambigüedades de fase permanecen constantes en el tiempo $C_{\in N^i_{\Delta t}}=0$. Dado que la aplicación que se va a dar en el presente trabajo al procesado PPP es posicionamiento estático, las covarianzas asociadas al ruido en los distintos elementos del parámetro coordenadas también serán cero. En el caso del estado del reloj del satélite, dado que su variación no se puede modelar, el ruido asociado a este parámetro se considera ruido blanco. Por último, para el retardo cenital, se le asocia un ruido de proceso de paso aleatorio.

4.2.5 MODELOS DE CORRECCIÓN APLICADOS AL POSICIONAMIENTO DE PUNTO PRECISO.

En la descripción del sistema de posicionamiento por satélite GPS llevado a cabo en el capítulo 2, se mencionaron de manera somera los distintos tipos de errores que afectan a las medidas. Al emplear la estrategia PPP los diferentes errores han de ser estudiados minuciosamente, dado que pueden llegar a producir grandes efectos (incluso del orden de metros), con el fin de reducir al mínimo su influencia en los resultados. Así pues, al emplear los productos precisos con combinaciones libres de efecto ionosférico de observaciones de la portadora de fase, es imprescindible tener en cuenta estos errores que pueden ser obviados en las medidas de pseudodistancia y modelos diferenciales. Los errores asociados a distintos efectos pueden agruparse en tres grupos (Kouba y Héroux, 2000): efectos derivados de la actitud del satélite, efectos derivados del emplazamiento y compatibilidades.

4.2.5.1 EFECTOS DERIVADOS DE LA ACTITUD DEL SATÉLITE.

4.2.5.1.1 OFFSETS DE LA ANTENA DEL SATÉLITE.

Dado que los modelos orbitales de los satélites se refieren a su centro de masas, los productos precisos también se refieren a este punto, a diferencia del mensaje de navegación, cuya información es referida al centro de fase de la antena. Por lo tanto, es necesario conocer los desplazamientos del centro de fase respecto al centro de masas del satélite, y tener en cuenta la orientación del vector desplazamiento en el espacio. Actualmente se emplea un valor individual y absoluto para la corrección de la posición del centro de fase para cada satélite, de la misma forma que para cada modelo de antena geodésica en tierra. Esta información se encuentra disponible en los archivos ANTEX,

pudiendo ser descargados de la página http://igscb.jpl.nasa.gob/igscb/station/general/.

4.2.5.1.2 EFECTO WIND UP.

Debido a que las señales emitidas por los satélites GPS están polarizadas circularmente, la fase de la frecuencia de la portadora recibida por el observador depende de la orientación mutua entre las antenas del satélite y receptor. Una rotación de cualquiera de las antenas alrededor de su eje vertical provocará un cambio en la fase de la portadora de hasta un ciclo, o en distancia una longitud de onda. Para un receptor de coordenadas fijas, este efecto se debe al movimiento orbital del satélite. Al desplazarse éste en su órbita, necesita realizar ciertas maniobras con el fin de mantener sus paneles solares dispuestos en la posición que más luz solar capten. Este efecto es absorbido por la posición de la estación y por el reloj del receptor.

4.2.5.2 EFECTOS DERIVADOS DEL EMPLAZAMIENTO.

Como resultado de las fuerzas gravitatorias a las que se ve sometida, la Tierra sufre una serie de deformaciones en su superficie, causadas principalmente por el Sol, la Luna y el efecto inercial derivado del movimiento de rotación terrestre. Los efectos de deformación son casi periódicos, alcanzando varios centímetros. Al afectar estos desplazamientos a grandes zonas de terreno, tienden a cancelarse al emplear el posicionamiento en red en líneas base cortas, inferiores a los 100 km. Sin embargo, en posicionamiento absoluto, estos desplazamientos han de ser tenidos en cuenta.

4.2.5.2.1 CARGA DE MAREA TERRESTRE.

La Tierra, es una superficie sólida cuyos componentes le confieren una cierta elasticidad, de manera que se deforma a merced de las mismas fuerzas gravitatorias que generan las mareas oceánicas. El potencial gravitatorio que se obtiene al emplear armónicos esféricos, al tratar los efectos de Luna y Sol por separado, en función de la posición del punto P de la superficie de la Tierra a observar es (Xu, 2003):

$$W_{P} = GM \sum_{n=2}^{\infty} \frac{\rho^{n}}{r^{n+1}} [P_{n}(sen\varphi)P_{n}(sen\delta) + 2\sum_{k=1}^{n} \frac{(n-k)!}{(n+k)!} P_{nk}(sen\varphi)P_{nk}(sen\delta)\cos(kH)]$$
[4.9]

donde φ es la latitud geocéntrica del punto P, δ es la declinación, H el ángulo horario local del astro y $P_{nk}(x)$ es el polinomio asociado de Legendre de grado n y orden k. El potencial de marea total se obtiene sumando los potenciales generados por Sol y Luna, resultando los desplazamientos de la corteza terrestre de ese potencial, en las direcciones radial, Norte y Este:

$$\Delta S_r = h \frac{W_p}{g} = \sum_{n=2}^{\infty} h_n \frac{W_p(n)}{g}$$
 [4.10]

$$\Delta S_n = l \frac{\partial W_p}{g \partial \varphi} = \sum_{n=2}^{\infty} l_n \frac{\partial W_P(n)}{g \partial \varphi}$$
 [4.11]

$$\Delta S_e = l \frac{\partial W_p}{g \cos \varphi \partial \lambda} = \sum_{n=2}^{\infty} l_n \frac{\partial W_P(n)}{g \cos \varphi \partial \lambda}$$
 [4.12]

donde g es la aceleración de la gravedad, λ es la longitud y h, l son los números de Love y Shida, números que aparecen como consecuencia de que la Tierra no es un cuerpo rígido ni tampoco perfectamente elástico.

4.2.5.2.2 MAREA POLAR.

Se conoce como marea polar a la respuesta elástica de la corteza terrestre generada por los desplazamientos del eje de rotación de la Tierra. Al igual que la carga de marea terrestre, estos movimientos también son periódicos y son debidos a pequeños cambios en el potencial centrífugo terrestre. La amplitud máxima del movimiento del polo de rotación sobre la superficie terrestre es de unas decenas de metros, describiendo una trayectoria que se corresponde con una elipse irregular, que se recorre en sentido levógiro, con un periodo de 7 años, que se puede descomponer en un periodo de 435 días (conocido como periodo de Chandler), con una amplitud variable de 0,1 s de arco y con variaciones cíclicas debidas a efectos estacionales. Utilizando los números de Love y Shida de grado 2, las correcciones expresadas en mm en las componentes radial, N y E son:

$$S_r = -32sen2\theta(m_1cos\lambda + m_2sen\lambda)$$
 [4.13]

$$S_N = -9cos2\theta(m_1cos\lambda + m_2sen\lambda) \qquad [4.14]$$

$$S_E = 9\cos\theta(m_1 sen\lambda - m_2 cos\lambda)$$
 [4.15]

donde m_1 y m_2 describen el desplazamiento diario del polo de rotación instantáneo respecto del polo medio en segundos de arco. Esta información se facilita mensualmente en el boletín B del IERS.

4.2.5.2.3 CARGA DE MAREA OCEÁNICA.

Se define marea como el movimiento vertical del nivel del mar, que dependiendo de la zona del planeta, pueden alcanzar varios metros. Este desplazamiento en la masa de agua de debe a los efectos gravitacionales provocados principalmente por Sol y Luna. La deformación por carga oceánica es similar a la que provocan las mareas terrestres, en el sentido en que dominan los periodos diurnos y semi-diurnos. Esta carga de marea es casi un orden de magnitud menor que las mareas terrestres. El efecto de la carga de marea oceánica puede despreciarse para posicionamientos sobre periodos de 24 horas en estaciones alejadas de los océanos, obteniendo precisiones de 5 cm. Sin embargo, en el posicionamiento absoluto por PPP, este efecto debe ser tenido en cuenta, a no ser que la estación se encuentre a más de 1000 km. de la línea de costa. Esto se debe a que los efectos de las mareas se reflejan en variaciones atmosféricas, con la consecuente variación en los parámetros de la troposfera y en la posición de la estación.

Los efectos de carga de marea oceánica pueden ser modelados en cada una de las principales direcciones mediante la siguiente corrección (IERS 2010):

$$\Delta c = \sum_{j} f_{j} A_{cj} \cos(\chi_{j}(t) + u_{j} - \phi_{cj})$$
 [4.16]

donde la amplitud A_{ci} y la fase ϕ_{ci} describen la respuesta del emplazamiento seleccionado a la carga de marea oceánica y son calculadas mediante convoluciones de funciones de Green utilizando los últimos modelos globales de mareas oceánicas. f_i y u_i dependen de la longitud del nodo lunar, pudiendo asumirse como 1 y 0 respectivamente para precisiones de 1 a 3 mm. El sumatorio en j representa los 11 armónicos de ondas de marea, 4 para las ondas semidiurnas, 4 para las ondas diurnas y 3 para las ondas de largo periodo. Los datos de amplitudes y fases para estaciones con coordenadas ITRF, con objeto de calcular el efecto de carga oceánica, se obtienen de servicios automatizados vía Internet, empleándose para el presente trabajo la página de loading provider web ocean tide http://www.oso.chalmers.se/~loading/. Tras el envío de un email con las coordenadas de la estación, envían un archivo en formato .BLQ, que es leído por el software de procesamiento, que en este caso será el Gipsy-Oasis.

4.2.5.2.4 PARÁMETROS DE ORIENTACIÓN DE LA TIERRA.

Los parámetros de orientación de la Tierra son necesarios para obtener la transformación entre los marcos de referencia celeste y terrestre. Estos parámetros son: las coordenadas del polo Norte terrestre instantáneo, la diferencia de tiempo UT1-UTC y las convenciones para el tiempo sidéreo,

precesión y nutación. La variación sub-diurna de estos parámetros es importante para alcanzar precisiones centimétricas en posicionamiento absoluto. Los cambios en las posiciones ITRF por efecto de la variación sub-diurna pueden alcanzar hasta 3 cm, en periodos diurnos o sub-diurnos asociados a mareas oceánicas. Por lo tanto, en posicionamiento estático de precisión sobre intervalos más cortos que 24 horas, este efecto sub-diurno debe ser tomado en cuenta para usuarios de órbitas precisas dadas en ITRF. Este efecto se puede modelar, al igual que se hace con los desplazamientos por marea, como correcciones aparentes a las coordenadas ITRF convencionales de la estación, evaluadas a partir de las correcciones instantáneas sub-diurnas de los parámetros de orientación de la Tierra.

4.2.5.3 COMPATIBILIDADES.

Para obtener los mejores resultados a la hora de emplear un análisis donde se constriña o fije parte de la información, como puede ser el empleo de un producto externo (como productos precisos), ha de aplicarse un conjunto de correcciones consistente, modelos, convenciones, estrategias de ponderación para órbitas y relojes, especialmente cuando se utilizan como referencia soluciones fijadas a un marco global. Es imprescindible ser consistente en aspectos como los productos precisos y la versión del ITRF, relojes y parámetros de orientación de la Tierra, calibración de antena, modelos troposféricos etc... La consistencia entre la información orbital, parámetros de orientación de la Tierra y correcciones del reloj de los satélites es el aspecto más importante a la hora de alcanzar la mayor precisión posible con PPP. Todo esto obliga a usar información de la misma fuente, evitando la mezcla de parámetros de distintos centros de análisis, debiendo, además, garantizarse que los modelos empleados por los centros de análisis para la estimación de órbitas, correcciones de los relojes y parámetros de orientación de la Tierra son los mismos que los implementados por el software de posicionamiento. Por último, en PPP han de tenerse en cuenta correcciones que con otras estrategias (como con dobles diferencias), como el wind up son omitidos, y como se ha mencionado con anterioridad, afectan de manera significativa a los resultados.

4.3 Estrategia de cálculo.

En líneas generales, la estrategia PPP requiere una carga de procesado mucho menor que los métodos diferenciales. Para realizar el cálculo de la posición, primero se pre-procesan las observaciones no diferenciadas de datos GPS. Estas observaciones se ajustan a los parámetros de observación fijados, como son el ratio de captura de datos o la máscara de elevación. El pre-

procesado de las observaciones no diferenciadas tiene como objeto detectar y eliminar las pérdidas de ciclo y posiciones fuera de rango (outliers).

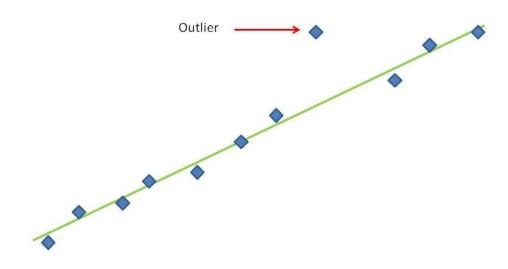


Fig. 4.1: Medidas fuera de rango.

Para la estimación de parámetros, i.e. coordenadas de la estación, correcciones al reloj del receptor y zpd, es necesario disponer de cierta información a-priori, como son unas coordenadas aproximadas de la estación, parámetros de orientación de la Tierra y cargas de marea oceánica y terrestre entre otros, además de las observaciones no diferenciadas. Al emplear PPP, en el procesamiento se emplean las observaciones de fase y de código simultáneamente, para formar la combinación ionosfera libre L3. Para el procesado se emplean, además de la combinación de observaciones ionosfera libre, los productos precisos, parámetros de orientación de la Tierra, correcciones por carga de marea oceánica y modelos ionosféricos globales con resolución de dos horas.

La estimación de los parámetros se desarrolla en tres pasos:

Primero se estiman, empleando las observaciones pre-procesadas, las coordenadas de la estación, correcciones al reloj del receptor y los retardos cenitales. En el caso de las coordenadas, la estimación es un valor medio para toda la sesión de observación. Las correcciones del reloj del receptor se estiman por época de observación y los retardos cenitales troposféricos se estiman como

- promedios para valores de una a dos horas. Los parámetros troposféricos y los residuales de las observaciones que se obtienen, se almacenan, ya que serán empleados en pasos sucesivos.
- Usando los residuos del primer paso, las observaciones se comprueban, buscando errores tales como saltos de ciclo y posiciones fuera de rango. Todavía en este punto, los saltos de ciclo se pueden conservar debido a que en el procesamiento de la fase, solo se discrimina los datos en el nivel del metro. A continuación se establece un primer umbral (por ejemplo 0,1 metros), y se repite el procesamiento de manera similar, tomando ahora los parámetros troposféricos como conocidos, es decir, no se estiman.
- Se examinan de nuevo las observaciones con un umbral para las residuales más preciso. Tras ser depuradas, las observaciones pasan el mismo proceso de la etapa anterior para realizar la estimación de parámetros. En el caso de que la solución vaya a ser combinada con otras sesiones, de distintos días por ejemplo, se almacenas las ecuaciones normales para los parámetros. Las coordenadas estimadas para la estación serán consistentes con el marco de referencia de los productos precisos empleados, siendo necesaria una transformación de 7 parámetros si se quieren expresar en una materialización específica del ITRF determinado.

4.3.1 CONVERGENCIA DE LA SOLUCIÓN PPP.

Las soluciones que se obtienen empleando el método PPP, convergen en función del tiempo dependiendo de los parámetros de varianza a priori y de la correlación de las observaciones no diferenciadas de código y fase. Así pues, la primera estimación se hace a partir de las observaciones de pseudodistancias, ya que se desconocen las ambigüedades de fase, de manera que la calidad de la posición la materializa la precisión de las observaciones de código. A medida que las observaciones de fase se añaden a la solución, las ambigüedades ionosfera libre y las coordenadas de la estación convergen a valores constantes, mientras que los parámetros cenitales de retardo troposférico y los parámetros de error del reloj del receptor varían en función a su ruido de proceso asignado.

CAPÍTULO 5. GIPSY-OASIS.

GIPSY-OASIS es un software de posicionamiento científico desarrollado por el JPL de la NASA, que en un principio se desarrolló como aplicación de posicionamiento relativo, evolucionando en el tiempo e incluyendo aplicaciones de posicionamiento absoluto. El software consta de dos partes con módulos en común. Por un lado se encuentra Gipsy (GPS-inferred Positioning System), empleado en aplicaciones geodésicas estándar, mientras la segunda parte, Oasis (Orbit Analysis Simulation Software), es un paquete dedicado al análisis de la covarianza de datos orbitales. El JPL comenzó a desarrollar el software en 1985, y ha estado en continuo desarrollo y mejora desde entonces. La versión actual (6.3), está programado en el lenguaje Perl, y corre sobre el sistema operativo Linux y se emplea no solo para datos GPS, también puede ser empleado con observaciones SLR (Satellite Laser Ranging), sobre satélites tipo TOPEX-POSEIDON (TOPography Experiment) y para órbitas de los satélites involucrados en el sistema DORIS (Doppler Orbitography and Radio positioning Integrated by Satellite).

5.1 Funcionamiento en red.

Gipsy-Oasis, a diferencia de otros softwares de procesamiento preciso, tiene la opción de trabajar sin el uso de dobles diferencias, lo que reduce de manera significativa la carga de procesado. Para poder trabajar con Gipsy, al igual que en los demás software científicos, los datos de entrada deben tener un determinado formato, llamado RINEX (Receiver INdependent Exchange). Cada receptor guarda los datos brutos de las observaciones en un formato propio, dependiente generalmente de la casa comercial que lo produce. Según esto, el primer paso será convertir los datos brutos de las observaciones a formato RINEX. Para ello, existen una serie de programas, en su mayoría libres, que facilitan esta conversión. Para la aplicación práctica que se desarrolla en el capítulo 6, el programa empleado es el TEQC de UNAVCO, que se detallará más adelante.

Una vez convertidos los datos brutos a formato RINEX, se procesan, pasando por una serie de etapas que involucran distintos programas para llegar a la solución con la precisión deseada. Las distintas etapas se muestran en la figura 5.1, que representa el diagrama de flujo de funcionamiento de Gipsy.

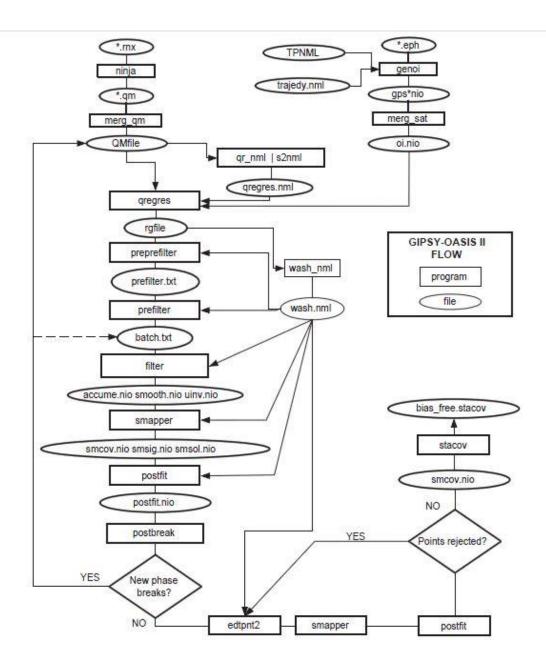


Fig. 5.1: Diagrama de flujo del procesado de datos en Gipsy-Oasis. Fuente: Thierry Gregorius 1996 .

En el diagrama de la figura 5.1 hay que diferenciar los archivos de datos, de los programas, representándose por óvalos y rectángulos respectivamente. A continuación se describen los programas y archivos involucrados en el procesado.

5.1.1 Ninja.

El primer programa en intervenir en el proceso es el llamado Ninja. Tiene como función leer el archivo RINEX, reordenándolo y traduciéndolo a formato binario. Acto seguido, el fichero RINEX traducido se envía a un sub-módulo llamado turbo-edit, que se encarga de borrar los errores groseros, tales como las posiciones fuera de rango, reparando, además, las discontinuidades en la señal en un 99%. El sub-módulo anterior es independiente de las órbitas de los satélites y la geometría de la constelación, de los estados de los relojes de satélite y receptor, de la troposfera y del movimiento de la antena (lo que permite que se use para aplicaciones cinemáticas). Después los datos son guardados en archivos binarios, clasificados para cada satélite y agrupado en archivos con extensión .qm, que son los ficheros que genera este programa.

5.1.2 Merg_qm.

La misión de este programa es leer la lista de ficheros con extensión .qm y agruparlos en un solo archivo llamado QMfile. Con objeto de crear una lista que pueda ser empleada por el siguiente programa, dos scripts llamados qr_nml y s2nml entran en juego, creando la lista qregres.nml.

5.1.3 Genoi.

La otra rama del diagrama de flujo es la concerniente a los datos de las órbitas de los satélites. El programa Genoi crea los archivos de extensión .oi, que contiene los datos de posición, velocidad, aceleración y derivadas parciales como función del tiempo de los distintos satélites. Para ello usa otros dos scripts, eci y trajedy.

- Eci

Este script extrae las órbitas radiodifundidas del mensaje de navegación, dándoles extensión .eph, referidas a un sistema ECI (Earth Centered Inertial) y asignándole formato .eci para que sean leídas por el programa Genoi. Estas órbitas se emplearan como datos de entrada para el script trajedy. La información relativa a movimiento del polo, se incorpora con un archivo en formato .nml, extraído del boletín B del IERS.

 Trajedy: Integra las órbitas creando un fichero .oi que mejor se ajusta al fichero eci. El fichero oi contiene toda la información del estado del satélite, así como de sus aceleraciones y derivadas parciales respecto al tiempo. A continuación se agrupan los ficheros con extensión .oi empleando el programa merge_sat, similar al empleado con los ficheros de extensión .qm, obteniendo como resultado los ficheros de extensión .nio.

5.1.4 Qregres.

Este es el programa encargado de realizar una regresión a los modelos físicos para determinar los parámetros que determinan la localización del receptor, el movimiento de la Tierra y los estados de los relojes de transmisor y receptor. Este programa aplica dos modelos: relacionados con la Tierra (efectos de mareas, rotación, etc.) y de observación, que relacionan los datos registrados con la realidad de la estación. Las entradas para este programa son las listas previamente creadas con extensión .nml y .nio. La salida es un fichero llamado rgfile, que está preparado para el siguiente proceso de filtrado.

5.1.5 El filtrado.

En este proceso son varios los programas que intervienen. En primer lugar el programa preprefilter crea un archivo de entrada llamado prefilter.txt a partir del fichero rgfile, y será la entrada del programa prefilter. Este último programa permite al usuario crear el archivo que contiene los eventos aleatorios tales como el ruido blanco del reloj, efectos troposféricos etc. Generando un archivo de salida batch.txt en el que los datos están ordenados de manera que dependen de los eventos aleatorios que ocurrieron al mismo tiempo a un grupo de datos. Por último, el programa filter realiza el filtrado como tal de los datos, empleando un algoritmo SRIF (Square Root Information Filter). Este filtro es numéricamente estable, ya que es una modificación de un filtro Kalman empleado para conocer la posición y velocidad de un objeto que se encuentra en un sistema relacionado con el tiempo, y que evita invertir grandes matrices de datos. El programa revisa el fichero rgfile en busca de ruido aleatorio en un evento específico y lo corrige. Filter produce hasta 5 tipos distintos de ficheros con extensión .nio (los más empleados son: accume.io, smooth.nio y uinv.nio), que contienen los resultados del filtrado, coeficientes de suavizado y las matrices de transición de estado.

5.1.6 Smapper.

Una vez que los datos han sido filtrados, se emplea el programa smapper, que calcula y mapea la covarianza, la sensibilidad y la solución de los parámetros del proceso. La salida de este programa son cuatro archivos, a saber, smsol, niosmcov, niosmsig, con extensión *.nio,* y APVALSOUT, que son la entrada para el proceso siguiente, llamado postfit. Este programa, junto con el filter, permite tratar cualquier parámetro como un parámetro estocástico.

5.1.7 Posfit.

Posfit es un programa cuya entrada se puede componer de hasta 4 ficheros, que son smsol.nio, uinv.nio, rgfile y un fichero de texto que contiene los pesos de las observaciones, máscara de elevación etc. y genera dos ficheros de salida, uno de puntos que contiene datos de las estaciones, llamado point.txt y otro de pre-ajuste que lleva información de los residuales, llamado postfit.nio.

5.1.8 Postbreak.

El programa postbreak lee el archivo postfit.nio en busca de discontinuidades en los residuales que el programa Ninja hubiese pasado por alto. En caso de encontrarse alguna discontinuidad, se modifica el archivo QM y se vuelve a ejecutar todo el proceso por medio del programa qregres. Cuando sólo se tienen errores por órbitas y no por estaciones terrestres, es suficiente modificar el archivo batch.txt y ejecutar filter, smapper y postfit.

5.1.9 Edtpnt2.

Este programa agrega y borra datos de puntos de la solución filtrada para poder actualizar los filtros y remover los valores fuera de rango. Una vez se ha ejecutado vuelven a correr los programas smapper y postfit, que evalúan los residuos y deciden si tiene que desechar algún punto. En caso de que esto ocurra, el archivo postfit.noise se modifica, y se vuelven a ejecutar de nuevo el smapper y postfit. En caso de no eliminarse ningún punto, se continúa con el proceso.

5.1.10 Stacov.

La herramienta Stacov incluye los siguientes sub-programas: Stacov, Heightfix, Stamrg, Statistics, y Transform. Stacov extrae las coordenadas de la estación del fichero smcov.nio, y las traduce de manera que el usuario pueda leerlas, mientras que heightfix es el encargado de actualizar la altura de la

antena para cada día de observación. Por su parte, el programa stamrg hace un ajuste de mínimos cuadrados entre las posiciones presentadas en diferentes archivos stacov, con la opción de determinar la velocidad de la estación a partir de las soluciones obtenidas para diferentes días. Statistics es el encargado de calcular las coordenadas geográficas (con la opción de presentar las series temporales) y las componentes para cada línea base, así como su equivalencia en distintos sistemas coordenados. Por último, transform se encarga de la transformación de un sistema de referencia a otro. Stacov produce como resultado un fichero de salida en formato de texto que contiene las coordenadas cartesianas de las estaciones y sus correspondientes errores.

5.3 Gipsy en PPP. Gd2p.pl

El software Gipsy-Oasis está en constante desarrollo y evolución. Como consecuencia de ello, a finales de 2008, aparece el programa gd2p.pl. Ésta es, en la actualidad, la herramienta principal para el posicionamiento de punto preciso con el que cuenta el software. El programa, escrito en el lenguaje Perl, funciona como un módulo más, y puede ser empleado tanto en posicionamiento estático como cinemático. La salida que da gd2.pl es la solución, que además de la posición, genera un fichero tdp_final, con los parámetros de error del reloj del receptor, parámetros troposféricos (parte seca y parte húmeda) y errores en la determinación de las fases con cada uno de los satélites desde el correspondiente receptor. También se otros tres archivos: un script ejecutable llamado run_again, que permite reprocesar los datos permitiendo al usuario mantener o cambiar la configuración del programa, los errores eventuales que se almacenan en el fichero gd2p.err, y por último el fichero gd2p.log que facilita un histórico de qué rutinas de GOA se han ejecutado.

5.3.1 Ficheros de estación.

Para poder ejecutar el programa gd2p.pl, además de los ficheros RINEX, son necesarios una serie de ficheros que proporcionan información acerca de la estación a procesar, que son almacenados en una carpeta llamada sta_info. Estos archivos son tres: sta_id, sta_pos y sta_svec.

Sta_id: Es el identificador de la estación, y debe figurar en mayúsculas. Presenta el siguiente formato: (espacio) ESTACIÓN (5 espacios) 0 (espacio) ESTACIÓN, e.g. ALBO 0 ALBO. Es fundamental que los espacios se correspondan con el formato, y la estación esté en mayúscula, ya que de lo contrario GOA no leería el fichero.

- Sta_pos: Fichero que contiene las coordenadas nominales de la estación, y que presenta el siguiente formato: (espacio) Estación (espacio) YYYY (espacio) MM (espacio) DD (espacio) HH (espacio) mm (espacio) SS.SS (espacio) (coordenada x en metros) (coordenada y en metros) (coordenada z en metros) (velocidad en la dirección x en metros/año, cero en caso de ser desconocida) (velocidad en la dirección y en metros/año, cero en caso de ser desconocida) (velocidad en la dirección z en metros/año, cero en caso de ser desconocida). YYYY es el año, MM es mes, DD el día, HH la hora, mm es el minuto, SS.SS es el segundo. Nuevamente, el mantener el formato es de vital importancia para evitar errores al ejecutar el programa. Las posiciones cartesianas se obtienen del fichero RINEX. Para ello, se emplea el comando *rnx2sta_pos* que extrae estos datos directamente, teniendo el fichero sta_pos sólo una línea por estación
- Sta svec: Con este fichero se introduce el offset de la antena. El formato de este fichero es el que sigue: (espacio) ESTACIÓN (YYYY) (espacio) (MM) (espacio) DD (espacio) HH:mm:SS.SS (espacio) (número de segundos totales de la observación) (código clasificador de la antena empeleada) (11 espacios) (velocidad en x) (10 espacios) (velocidad en y) (10 espacios) (velocidad en z) (5 espacios) (altura vertical de la antena) c (comentarios). Pueden existir varios ficheros sta_svec en el lugar en el que se almacenen, de manera que cada fichero represente un cambio en la antena (por ejemplo, para el caso de la isla de Alborán, cambios en la antena debidos a reparaciones por la corrosión). GOA usará para el procesado el último de los ficheros sta_svec, por lo que a la hora de incluirlo en la carpeta sta_info, esto último ha de tenerse en cuenta. Los códigos con los que se clasifican las antenas pueden encontrarse en: /goa/sta_info/pcenter, y en el caso de que se desconozca la velocidad, se le atribuye un valor de 0.

De forma resumida, el contenido de estos tres ficheros facilita un identificador único para cada estación, su posición aproximada (que sirve de posición inicial en los cálculos) y la excentricidad del punto a posicionar, con respecto al centro de fase de la antena para que el punto sea siempre el mismo aunque se cambie la antena o la base en la que se asienta. También es imprescindible que la antena del receptor a posicionar, tal como se ha descrito en el fichero svec, aparezca definida en el fichero pcenter. Existe la posibilidad de obtener la información para los tres ficheros anteriores extrayéndolos directamente del fichero RINEX mediante los comandos rnx2sta_id, rnx2sta_pos y rnx2sta_svec. El empleo de los comandos anteriores facilita mucho el trabajo, ya que los ficheros de la estación tienen un formato determinado, donde se conservan los espacios y se diferencias los caracteres en mayúsculas y minúsculas.

5.3.2 Preparación para el procesado.

Para el presente trabajo se han creado una serie de scripts, que preparan las observaciones para que el software pueda reconocerlas y tratarlas al ejecutarse el comando gd2.pl. En primer lugar se fija la variable de ambiente "year". Con esto, lo que se busca es procesar las observaciones de una determinada estación correspondientes a un cierto año . Seguidamente, el script "crea_angel" se encarga de convertir dos fechas seleccionadas en variables de ambiente conocidas, bajo los nombres doy1 y doy2, y cuyo significado es day of year. Se trata de convertir una fecha concreta en el valor numérico asociado a un día dentro del año fijado. Así pues, si se escogen como fechas los días 31 de enero y 4 de febrero, el resultado será: doy1= 31 y doy2=35. Este script crea dos variables por un motivo, para evitar el procesar todo un año completo, y poder procesar sólo las observaciones comprendidas entre dos fechas cualesquiera en caso de que sea necesario.

El siguiente script es el llamado solo_angel. En este script se debe fijar primeramente la variable de ambiente year, y escoger las dos fechas entre las que se quiere procesar datos. A continuación se selecciona la estación deseada, y en caso de que no esté procesada previamente, se crea una carpeta en la que se almacenarán los datos del procesado. Este script a su vez llama al script solo_new_angel, en el que se han introducido las líneas de código correspondientes al comando gd2.pl. Además, también se llaman a los scripts gipsy2xyz y gipsy2gd, que proporciona las coordenadas de la estación procesada en coordenadas cartesianas geocéntricas y en coordenadas geodésicas, para cada día a las 12 horas UTC. Estas coordenadas se extraen del fichero stacov, y se encuentran ya en el marco de referencia IGS08.

En el script solo_new_angel es dónde se encuentra la llamada al programa gd2.pl. Como se ha dicho, este script es llamado a su vez por el script solo_angel. Durante su ejecución, otros varios scripts son llamados, para adecuar el formato de las observaciones. Por ejemplo, el primer script llamada es el year2cal, cuya misión es convertir la variable year, en un número de dos dígitos, de manera que si tratamos de procesar el año 2014, al ejecutarse el script anterior, el número que devolverá será 14, creando una variable de ambiente con ese valor. Tanto este como otros scripts llamados durante este proceso, vienen incluidos en el software GOA. El objeto de tener que realizar todo esto antes de llegar a la ejecución del principal programa para PPP, no es otro que el de adaptar tanto la fecha como el nombre de la estación, a partir de los datos brutos de la observación, a un formato legible por el software. Esto es debido a que, al trabajar en Linux, cualquier error de formato produce errores al ejecutarse, debiéndose mantener, por ejemplo, los espacios entre líneas y las letras en mayúsculas o minúsculas.

El fichero *gd2p.err* que se crea es en realidad un histórico del proceso que se lleva a cabo. Este fichero es de gran utilidad llegado el momento de una interrupción en el procesado, siendo éste el archivo a consultar en caso de error, para ver qué ha sucedido.

Dentro de la preparación de los datos para el procesado, es necesario contar con los datos referentes a las efemérides de los satélites. Para ello, y de forma automática (mediante un script creado a tal efecto), todas las semanas se descargan vía FTP las efemérides precisas del JPL (Jet Propulsion Laboratory). Estas efemérides vienen comprimidas, de manera que antes de continuar deben ser descomprimidas. Una vez hecho esto, y para cada día, el script se encarga de coger la información necesaria para el procesado. Así mismo, también es necesarios extraer de la carpeta de datos de la estación los parámetros referidos a la antena que emplea el receptor.

Un paso de gran importancia, llegados a este punto, es la conversión de los datos de observación brutos a formato RINEX. Para ello, dentro de la rutina del script solo_new_angel se incluye una llamada al programa TEQC de UNAVCO. Esta es una herramienta empleada en pre-procesado de datos GNSS, y cuyo principal cometido es la traducción de un fichero de observaciones bruto (es decir, ficheros en el formato nativo de cada casa comercial) a otros formatos. No sólo permite la traducción, además también es posible la extracción de ciertos datos y la edición o corrección. El receptor y antena empleados para la aplicación práctica son de la marca Trimble, y serán descritos en el siguiente capítulo.

5.3.3 Gd2p.pl

Llegados a este punto, y con las observaciones ya preparadas, se ejecuta el programa con el que cuenta GOA para el posicionamiento de punto preciso: Gd2p.pl. Para ejecutarlo es necesario contar con un dato más, que no se ha obtenido durante el proceso, la carga de marea oceánica. Para conseguir esta información, se recurre al sitio web http://holt.oso.chalmers.se/loading/. Una vez ahí, se introduce una posición aproximada de la estación y una dirección de correo a donde se envían, en un corto plazo de tiempo, los parámetros de carga de marea oceánica, que deben ser almacenados en la carpeta que contiene la información de las estaciones, en la que GOA buscará estos datos en caso de que se configure en este sentido.

Junto con los parámetros anteriormente mencionados, es necesario introducir una serie de parámetros, que son extraídos de las efemérides precisas. Un ejemplo de esto último es el decimado de las observaciones, y se escoge como opción dentro del programa *gd2p.pl*. En este caso se ha escogido un decimado de 300 segundos, que es el intervalo de tiempo que emplean las efemérides precisas del JPL para dar la posición de los satélites. Al hacer esto, se evita que el software interpole entre dos instantes de tiempo para hallar las

coordenadas cartesianas de los satélites. Otro parámetro que entra en juego en esta parte del procesado es el diagrama de radiación de la antena, que puede ser extraido de un fichero sinex teniendo la ID de la estación y el fichero antex del IGS, mediante la subrutina sta2antxyz.py, que debe ser preparado con anterioridad y depende del sistema de referencia en el que se esté trabajando.

Por último, se ejecutan los módulos *qregres* y *stacov*. El objetivo de emplear el módulo *qregres* no es otro que la aplicación de los modelos físicos para la ubicación y movimiento de la estación y de la Tierra, como pueden ser la rotación o el movimiento del polo. Por otra parte, el módulo *stacov* extrae las coordenadas resultantes del proceso, y las convierte del código binario de origen, a código ASCII, para que el usuario pueda leerlas.

El resultado final del proceso, es una carpeta, llamada de la misma forma que la ID de la estación, con una sub-carpeta por cada día de observación. Estas sub-carpetas irán numeradas en función del día dentro del año que ocupe la fecha de las observaciones. En la carpeta principal aparecen una serie de ficheros comprimidos, que dan la siguiente información:

- YYYY-MM-DD.ant.gz: Identifica los ficheros de calibración de antena de emisor y receptor empleados para generar los estados del los relojes y las órbitas.
- YYYY-MM-DD.frame.gz: Indica el marco en el que se proveen las órbitas.
- YYYY-MM-DD..nf.eo.gz: Parámetros de orientación de la Tierra empleados en la solución fiducial libre.
- YYYY-MM-DD.nf.pos.gz: Posiciones cartesianas geocéntricas de las órbitas de todos los satélites empleados en la solución libre fiducial.
- YYYY-MM-DD.nf.tdp.gz: Estados de los relojes y ratio de variación de actitud de todos los satélites cada 5 minutos para la solución libre fiducial.
- YYYY-MM-DD.nf.wlpb.gz: Conjunto de señales globales, que se utilizan en la determinación de las ambigüedades.
- YYYY-MM-DD.shad.gz: Eventos de sombra, es decir, instantes en los que el satélite entra y sale del cono de sombra de la Tierra. Mientras que está en esa zona no sabe cómo dirigir sus paneles hacia el Sol, y su actitud se vuelve inestable. Como resultado la antena no apunta al nadir, y no es posible determinar la actitud del satélite con calidad suficiente como para que las señales sean utilizables en un posicionamiento de precisión. Lo más práctico es eliminar del procesado todos los datos procedentes de satélites durante el intervalo de tiempo en el que se encuentra dentro del cono de sombra de la Tierra.

Además, también se obtiene un fichero con extensión .x que proporciona los parámetros de transformación (filas 4 a 10) para el paso del sistema de mínimos constreñimientos al IGS05 (ver figura 5.2). Las tres primeras

filas son los resultados estadísticos, y las demás dan los residuos de las posiciones y velocidades de las estaciones que intervienen en la fase núcleo del proceso con respecto a la propia transformación.

CHI^2	=	0.83	39E+02			
DOF	=		62			
CHI^2	= /DOF =		1.35			
RX =	3620E	2-09	+- 0.714	7E-10 ra	d	
RY =	0.1362	80-3	+- 0.772	6E-10 ra	d	
RZ =	2196E	-08	+- 0.755	6E-10 ra	d	
TX =	8445E	E-02 -	+- 0.102	0E-02 m		
TY =	0.1802E	E-03 -	+- 0.991	5E-03 m		
TZ =	5955E	E-02 -	+- 0.106	5E-02 m		
S =	0.6921	E-10 ·	+- 0.251	0E-09 pa	rts	
POSIT	ION DIF	FERENCI	ES AND ER	RORS IN M		
NAME	N	E	V	SN	SE	SV
ARTU	1.6	3.6	V 3.5 1.9	2.2	1.7	6.0
BRUS	-0.1	1.8	1.9	2.2	1.9	6.2
CEDU		-2.1	2.9	2.1	1.8	6.0
CHAT		1.3	-5.3	2.2	2.1	5.9
CONZ	4.2	-0.5	2.4	2.8	2.6	7.9
CR01	2.0	1.4	-0.6	2.0	2.2	8.0
DAEJ	1.4	-2 1	2.7	2.4	2.0	7.1
GLPS	1.9	-1.8	7.6 15.2 -0.3	2.0	2.2	6.4
GUAO	0.2	0.2	15.2	3.3		
KARR	3.8	0.8	-0.3	2.0	2.0	6.5
KIT3	0.6	0.4	0.6	2.9	2.6	11.2
KOKB		4.4	3.3	2.0	2.8 2.0 2.6 2.2	6.2
LPGS	-1.8	-1.9	13.7	3.1	2.8	11.9
MATE	1.2	0.0	-3.5	2.0	1.8	5.7
MD01	1.6	-0.4	-12.3	2.0	1.9	5.8
NRIL		2.2	-2.9	2.0	1.7	6.1
QAQ1		-0.5	3.0	2.5	2.0	7.2
SEY1		-6.3	6.8	2.7	2.9	12.5
SYOG			3.5	2.4		
THTI	2.0	-1.8	12.1	2.1	2.4	6.5
TRO1	-1.6	-1.6	-4.6	1.8	1.6	5.6
ULAB			-8.5	2.4		
WILL	-1.5	0.1	-10.9	2.2	1.7	5.2
WRMS	2.1	2.0	6.7			

Fig. 5.2: Fichero con los parámetros de la transformación el 1 de marzo de 2009.

En el fichero con los parámetros de transformación aparecen los valores para las tres rotaciones, tres traslaciones y el factor de escala que llevan del sistema de mínimos constreñimientos (cada día se usa un sistema de referencia no constreñido) al marco de referencia terrestre internacional. También indica con cuantos grados de libertad (función del número de estaciones utilizado) se han calculado, así como las estaciones que se han empleado para realizar dicho cálculo, con sus diferencias en coordenadas y velocidades respecto la posición de cada una de ellas. Por último, se obtienen también como resultado dos ficheros con extensión .dat, en el que se dan las

coordenadas cartesianas geocéntricas de cada estación, así como las geodésicas, ya transformadas en el sistema de referencia escogido.

En las distintas sub-carpetas que se van obteniendo para cada día procesado, se almacenan una serie de ficheros que van facilitando la posición en distintos sistemas. Así pues, en primer lugar tendremos un fichero con extensión .gd, en el que se almacena la posición del receptor en el sistema libre donde se calcula mediante PPP, en coordenadas geodésicas. Los ficheros con extensión _fix.gd contienen las coordenadas del receptor en el sistema libre con resolución de la ambigüedad con su correspondiente matriz de varianza-covarianza en coordenadas geodésicas. Por último, los ficheros con extensión _fix_tra.gd contienen la posición del receptor en coordenadas matriz correspondiente de geodésicas con su varianza-covarianza transformado al sistema de referencia escogido.

CAPÍTULO 6. Aplicación práctica a la geodinámica de la Isla de Alborán.

La Isla de Alborán isla de soberanía española es una (administrativamente depende de la provincia de Almería) ubicada al Oeste del Estrecho de Gibraltar (lat= 35.938279004 N long= 3.035814358 W), a 48 millas náuticas al Sur de Adra y 35 millas náuticas del cabo de Tres Forcas. Tiene una extensión de 71.200 metros cuadrados, con una longitud de 642 metros, 265 metros de ancho y una altura máxima de 16 metros. En la isla existen dos edificaciones: un faro, y un edificio auxiliar. Así mismo, cuenta con un embarcadero un helipuerto y un cementerio. La Armada Española mantiene un destacamento permanente en la isla, encargada de la custodia del faro y eventual ayuda a embarcaciones con problemas en la zona. Además, esta isla es un santuario de aves, guareciendo a dos especies protegidas.

A finales del año 2008, dada la importancia de su posición , se instala por parte del Real Observatorio de la Armada (con autorización y apoyo del entonces denominado Ministerio de Medio Ambiente, Medio Rural y Medio Marino) una estación permanente GPS, en el edificio del faro. La estación comienza a funcionar en el año 2009, manteniéndose en constante funcionamiento (excepto por una serie de saltos para efectuar mantenimientos, debido al deterioro que la estructura y las conexiones de la antena sufren al estar expuestos en un ambiente marino, especialmente agresivo dado lo habitual de los temporales de viento de la zona) desde entonces. Los datos brutos de las observaciones se envían diariamente al ROA vía intranet (Red de Propósito General del Ministerio de Defensa).



Fig. 6.1: Imagen satélite de la Isla de Alborán. Fuente: Google Maps.

6.1 Descripción de la región.

La situación de la Isla de Alborán la marca su ubicación respecto a dos grandes placas tectónicas, la Euro-asiática y la Africana. La zona de contacto de estas dos placas tiene una gran extensión, desde las islas Azores en su parte más occidental, hasta el Estrecho de Gibraltar, continuando por el Norte de África en dirección a Túnez. Alborán se encuentra al Sur de la zona subbética y las costas mediterráneas que van desde Estepona hasta Cabo de Palos, comprendiendo los terrenos superpuestos tectónicamente a las unidades de la cordillera Bética y del Rif. La estructura tectónica de la zona se caracteriza por la superposición de mantos de corrimiento cuyas traslaciones alcanzaron decenas de kilómetros (Perez Peña, 2007). Es en esta zona de contacto entre las dos placas, en la falla Azores-Gibraltar, a unos 200 km de Cabo San Vicente, donde en 1755 se produce un gran sismo, de una magnitud estimada de entre 8 y 9 grados en la escala de Ritcher, conocido como el Terremoto de Lisboa. Dicho sismo vino seguido por un tsunami que arrasó las ciudades de Huelva y Cádiz.

6.2 Descripción de los equipos.

Como se ha mencionado antes, el ROA instala un receptor GPS geodésico en el edificio del faro. El receptor es un modelo NETRS de la casa comercial Trimble. Este es un receptor bi-frecuencia que trabaja sobre un sistema operativo Linux, con la posibilidad de ser operado tanto en red, como de manera aislada. Por su parte, la antena es del tipo Choke Ring, empleando un diseño desarrollado en el JPL (Jet Propulsion Lab), y que es ampliamente aceptado por la comunidad científica. El envío de datos se realiza vía intranet de Defensa. Los datos son enviados por el receptor, sin necesidad de tener un pc conectado (éste solo se usa para configurar el receptor). A continuación se muestran algunas imágenes de los distintos componentes de la estación.





Fig. 6.2: Imágenes de la antena ubicada en el edificio del faro.



Fig. 6.3: Imagen del receptor y el ordenador.

El principal problema que afecta al equipo es el de la corrosión. Debido a que se encuentra en un ambiente marino, la salinidad es muy alta, lo que produce rápidos deterioros, no sólo en la estructura en la que va montada la antena, sino que también en el conector del cable de datos y en el perno de sujeción de la antena a la basada. Como consecuencia de ello, la estación estuvo fuera de servicio en cuatro ocasiones, teniendo en 2010 el mayor periodo sin datos de observaciones. Existen planes para la sustitución de la estructura donde va montada la antena por una de hormigón, si bien los problemas tanto en el conector como en el tornillo de sujeción no tienes una solución práctica, más allá de impregnarlos en vaselina o una sustancia oleosa.

6.3 Configuración del receptor.

El receptor NETRS de la casa Trimble puede ser configurado de múltiples formas, pudiendo ser ajustado para las necesidades de cada observación. Todos los ajustes necesarios se realizan a través de un pc conectado al receptor vía Intranet, que tiene una dirección IP y da acceso a una página de configuración y control del equipo, como se muestra en las figuras 6.4 y 6.5. Para el caso de Alborán, el receptor está programado para registrar sesiones de 60 minutos, con observaciones cada 30 segundos, y cero grados de máscara de elevación.

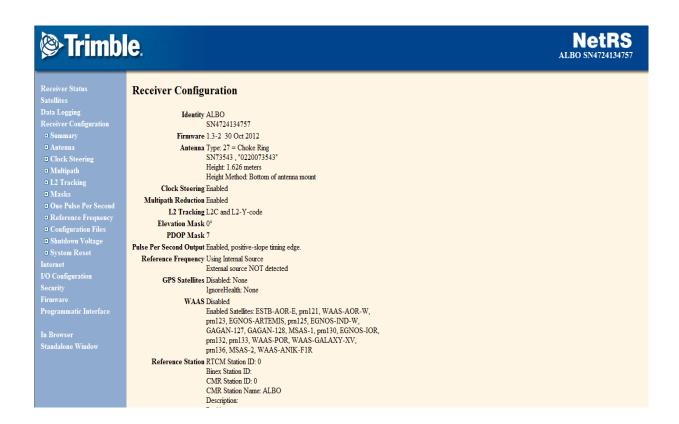


Fig. 6.4: Imagen de la página de configuración del receptor dónde muestra la configuración actual.

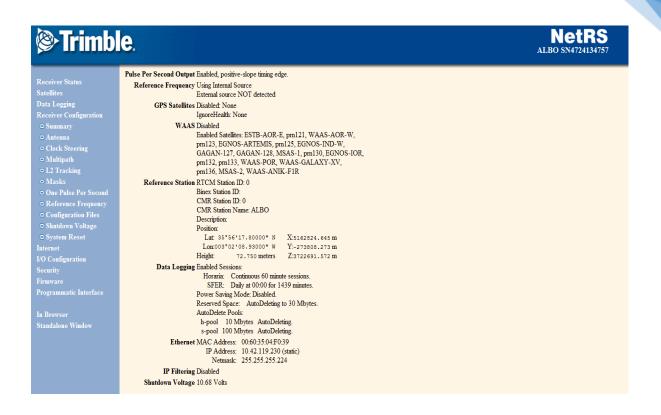


Fig. 6.5: Continuación de la figura 6.4.

Todos los días los (io del receptor y que se emplearan de back up) como los ticheros RINEX de las observaciones se envían vía Intranet de Defensa al Real Observatorio de la Armada, donde son almacenados. Dentro de la página que controla el receptor se pueden ver la posición de la estación, satélites en seguimiento y cualquier información necesaria relacionada con el equipo y las opciones de observación. También existe la posibilidad de cambiar el nombre de la estación, la dirección IP ver la memoria ocupada y libre que tiene el receptor o el tipo de antena asociado.

6.4 Procesado de los datos y conclusiones.

Para el procesado de los datos se emplean los scripts descritos en el capítulo anterior. Durante los 5 años de observaciones, y debido a los problemas derivados del ambientes y que ya han sido descritos, existen 4 intervalos de tiempo en el que no se dispone de registros. Estos periodos son los que siguen:

- Año 2009: Los registros comienzan el día 33 (2 de febrero), estando el resto del año completo.
- Año 2010: Los registros comienzan el día 4 (4 de enero), estando el resto del año completo.

- Año 2011: Los registros comienzan el día 3 (3 de enero), estando completos hasta el día 306 (7 de noviembre).
 - Año 2012: Los registros comienzan el día 148 (28 de mayo).
- Año 2013: Los registros comienzan el día 7 (7 de enero). Falta el día 182 (1 de julio) y el último día de registro es el 294 (21 de octubre).
- Año 2014: Primer día con registros es el 103 (14 de abril), estando completos hasta el día de hoy.

Como se puede apreciar, existen gaps importantes. Esto ha sido inevitable, dado que el desplazamiento a la isla, para realizar las correspondientes reparaciones no siempre ha sido posible, debido a lo aislado del emplazamiento, siendo necesario el traslado por helicóptero o por embarcación.

Los resultados para la serie temporal 2009-2014 en coordenadas geodésicas para las distintas épocas se muestran en los siguientes gráficos:

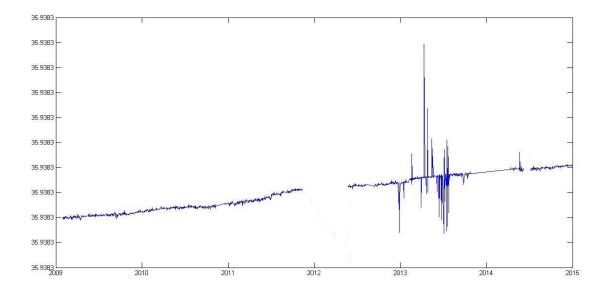


Fig. 6.6: Latitud.

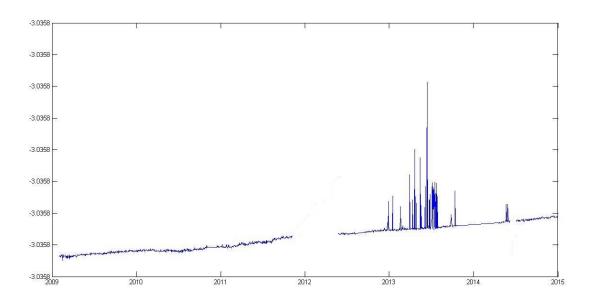


Fig. 6.7: Longitud.

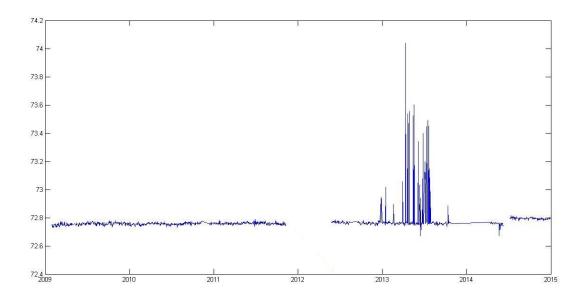


Fig. 6.8: Altura.

En los resultados obtenidos se puede apreciar perfectamente los periodos de inactividad de la estación. La serie temporal, además, está sin depurar, algo que se ve a simple vista, en forma de los errores groseros que afectan a los resultados. GOA dispone de una serie de herramientas para llevar a cabo este proceso. Para ello, se emplea el comando *stamrg*, que combina los fichero *stacov*, pudiendo emplear o no un modelo de velocidad a priori. Una vez empleado el comando *stamrg*, se ejecuta *statistics*, que como ya se ha mencionado devuelve las coordenadas reflejadas en el fichero *stacov* en forma de coordenadas geocéntricas, geodésicas, ltu o env baselines. Para el cálculo

de las coordenadas geodésicas se emplean los estándares IERS 2010. El filtrado de la serie se hace de forma manual, estableciendo un error de un centímetro. Hecho esto, el resultado será una serie temporal con tendencia, cuya pendiente, en primera aproximación, dará la velocidad de la estación y los saltos de serie.

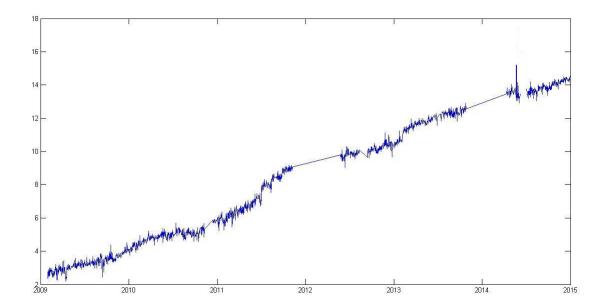


Fig. 6.11: Altura.

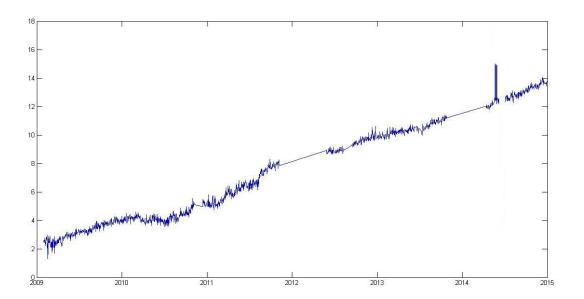


Fig. 6.10: Longitud.

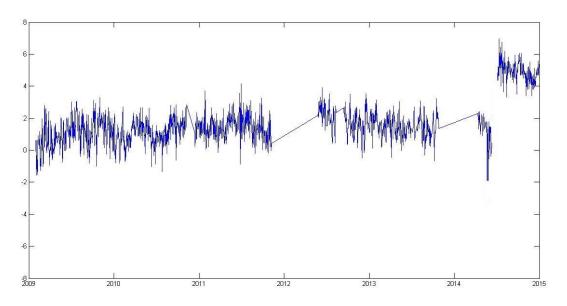


Fig. 6.11: Altura.

En los tres gráficos anteriores se muestran los resultados obtenidos. A la vista de dichos resultados, se evidencia un movimiento horizontal en el tiempo de la estación, que se puede descomponer en dos movimientos, uno en dirección E y otro en dirección N, lo que es coherente con un desplazamiento de las placas Euro-asiática y Africana.

6.5 Trabajos futuros.

La pendiente en la serie temporal con tendencia da en una primera aproximación una velocidad para la estación. La identificación del ruido de color que afecta al movimiento de la estación (qué color de ruido) para refinar el valor y los errores que afectan al valor de la velocidad, con el fin de generar parámetros de amplitud de los movimientos de la estación además de calcular las magnitudes de los saltos de la serie. Para ello, se emplearía el software CATS (Simon Williams, 2008). Con una velocidad refinada, se puede obtener un modelo de movimiento de la estación, que se ajuste a la siguiente fórmula:

$$x(t) = a + bt + \sum_{j=1}^{p} (A_j sen(w_j t) + B_j cos(w_j t)) + \sum_{j=1}^{p} C_j H(t - T_j)$$

donde:

- a es el valor inicial.
- b es la velocidad de la serie.
- w es la frecuencia angular de las componentes armónicas.
- C_i son las discontinuidades descritas por Heaviside.
- T_i son los instantes de tiempo de las discontinuidades.

El modelo de movimiento obtenido de la estación tiene un gran valor de cara al estudio del tipo de desplazamientos que afectan al emplazamiento, especialmente los periódicos, de forma que se puedan clasificar en forma del periodo que siguen, como, por ejemplo, movimientos diurnos o semi-diurnos.

Anexo I: Script *crea_angel*.

```
if ( $#argv < 2 ) then
  echo "Introduce dos valores de DOY"
     exit
endif
@ j = $1
@ jmax = $2
while ($j <= $jmax)
if ($j<10) then
  set jj = 00\$j
  else if (($j<100) && ($j>9)) then
  set jj = 0$j
  else
  set jj = \$j
endif
mkdir $jj
@ j++
end
exit
Anexo II: script solo_angel.
setenv year 2014
setenv doy1 1
setenv doy2 365
#foreach sta (acor alba bell cace casc ceu1 creu esco huel lago)
foreach sta (albo)
```

```
#si no existe la estacion la crea
 if! -r $sta then
 mkdir $sta
 endif
 cd $sta
 mkdir elresto
 cd elresto
 #proceso dias entre DOYs selecinados en crea_angel
 source /home/gps/prueba/script/crea_angel $doy1 $doy2
 source /home/gps/prueba/script/solo_new_angel
 cd ..
 source /home/gps/prueba/script/gipsy2xyz
 source /home/gps/prueba/script/gipsy2gd
 cd ..
end
Anexo III: script solo_new_angel.
#setenv anio {`/goa/bin/year2cal $year`}
unset j
unsetenv j
unsetenv doy
setenv anio {`$home/scripts/year2cal $year`}
foreach j (*)
 doy2cal {$anio} {$j} > diahoy
 source $home/scripts/formato_doy > dia_formato
 setenv doy `cut -b 1-10 dia_formato`
  setenv mm `cut -b 6-7 dia_formato`
  setenv dd `cut -b 9-10 dia_formato`
 rm diahoy dia_formato
```

```
unsetenv CC YY MM DD
if -r ../malos/$j/gd2p.err then
 rmdir $j
else
if -r ../$j/{$sta}tra.stacov then
rmdir $i
else
 if -r $home/datos/$year/$sta/{$sta}{$j}0.{$anio}d.gz then
                                $home/datos/$year/$sta/{$sta}{$j}0.{$anio}d.gz
$home/datos/$year/$sta/{$sta}{$j}0.{$anio}d.Z
 endif
 if -r $home/datos/$year/$sta/{$sta}{$j}0.{$anio}d.Z then
 if -r $home/datos/$year/orbitas/{$doy}_nf.wlpb.gz then
 cd $i
 echo $dd $mm
 rm -r *
 s2nml -info $GOA_STA_INFO -d $year $mm $dd -hv $sta > sta_datos
 echo $GOA_STA_INFO $sta
 echo "noooooooooo"
 grep 'Antenna type:' sta_datos > antena
 setenv ANTENA 'cut -b 19-27 antena'
 echo "lucas"
 echo $ANTENA
 echo "lucas"
 echo $sta > sta_min
 setenv STA `cat sta_min|tr '[a-z]' '[A-Z]'
 mkdir raw
 cd raw
 cp $home/datos/$year/$sta/{$sta}{$j}0.{$anio}d.Z.
```

```
$GOA/bin/crz2rnx {$sta}{$j}0.{$anio}d.Z
 rm {$sta}{$j}0.{$anio}d.Z
 /home/gps/teqc/teqc +obs {$sta}{$j}0.{$anio}a -R -O.obs "L1L2C1P2P1"
{$sta}{$j}0.{$anio}o
 mv {$sta}{$j}0.{$anio}a {$sta}{$j}0.{$anio}o
 grep -i NetRS {$sta}{$j}0.{$anio}o > recep.txt
 if (-r recep.txt) then
                     ' 'TRIMBLE NETRS' {$sta}{$j}0.{$anio}o
 replace 'NetRS
 endif
 grep -i POLARX3ETR {$sta}{$j}0.{$anio}o > recep.txt
 if (-r recep.txt) then
  replace 'POLARX3ETR' 'POLARX2 ' {$sta}{$j}0.{$anio}o
 endif
 grep -i GR10 {$sta}{$j}0.{$anio}o > recep.txt
 if (-r recep.txt) then
  replace 'GR10 ' 'GRX1200' {$sta}{$j}0.{$anio}o
 endif
 grep -i 'TPS NET-G3 ' {$sta}{$j}0.{$anio}o > recep.txt
 if (-r recep.txt) then
   replace 'TPS NET-G3' 'TPS NETG3' {$sta}{$j}0.{$anio}o
 endif
cd ..
 cp $home/datos/$year/orbitas/{$doy}*.gz .
 gzip -d {$doy}*.gz
 mv {$doy}*pos pos
 mv {$doy}*eo tpnml
 mv {$doy}*shad shadow
 mv {$doy}*tdp tdp_clk_yaw
 mv {$doy}*frame frame
 mv {$doy}*wlpb wlpb
```

```
mv {$doy}*ant GPS_antcal_ref
 mv {$doy}*.x transformation
 setenv FRAME `cut -b 1-5 frame`
 cp /var/opt/goa-var/sta_info/{$FRAME}/{$ANTENA}.xyz .
 replace NAME $STA {$ANTENA}.xyz
 gd2p.pl -i raw/{$sta}{$j}0.{$anio}o -goa_prod_here -n {$sta} -add_ocnld " -c
$GOA_STA_INFO/ocnId_coeffs" -r 300 -interp_300 -type s
                                                             -dwght 1.0E-5
1.0E-3 -d {$doy} -trop_z_rw 5E-8 -wetzgrad 5E-9 -w_elmin 3 -AntCal
{$ANTENA}.xyz -OcnldCpn -tides WahrK1 PolTid FreqDepLove OctTid -arp -
amb_res 1 > & gd2p.err
 stacov -i smcov.nio -o {$sta}.stacov -tp tpnml -q qregres.nml
 statistics -r {$sta}.stacov -gdr
 apply -x transformation -i {$sta}.stacov -o {$sta}tra.stacov -r -t -s -novel
 statistics -gdr -r {$sta}tra.stacov
 stacov -i smcov2.nio -o {$sta}_fix.stacov -tp tpnml -q gregres.nml
 statistics -r {$sta}_fix.stacov -gdr
 apply -x transformation -i {$sta}_fix.stacov -o {$sta}_fix_tra.stacov -r -t -s -
novel
 statistics -gdr -r {$sta}_fix_tra.stacov
 rm -r raw
 rm {$ANTENA}.xyz sta_datos sta_min
 rm rgfile batch* *nio q*nml qmfile
 rm pos tpnml shadow tdp_clk_yaw frame wlpb GPS_antcal_ref transformation
 rm *postbreak*
 rm was* postfit* *filter*
 unsetenv dd
 unsetenv doy
 unsetenv mm
 unsetenv STA
 unsetenv ANTENA
 pwd
```

```
cd ..

if -r ../$j/input.files then

rm -r ../$j

endif

mv $j ..

else rmdir $j

endif

else rmdir $j

endif

endif
```

Anexo IV: gipsy2gd.

```
set station=$sta

#rm ${sta}_gd.dat

#ano bisiesto

set ii = 365

if ((($year % 4 == 0) && ($year % 100 != 0)) || ($year % 400 == 0)) set ii = 366

foreach i (*)

cd $i

if -e ${station}_fix_tra.stacov then

# Cabecera del fichero de salida:

set lat=`awk 'NR==1 {print $4}' ${station}_fix_tra.gd`

set lon=`awk 'NR==2 {print $4}' ${station}_fix_tra.gd`

set rad=`awk 'NR==3 {print $4}' ${station}_fix_tra.gd`
```

```
echo $year $i $ii $lat $lon $rad | awk '{printf "%9.4f %s %s %s \n",$1+($2-0.5)/$3,$4,$5,$6}' >> ../${sta}_gd.dat
endif
cd ..
end
exit
```

Anexo V: gipsy2xyz.

exit

```
set station=$sta
#rm ${sta}_xyz.dat
#ano bisiesto
set ii = 365
if ((($year % 4 == 0) && ($year % 100 != 0)) || ($year % 400 == 0)) set ii = 366
foreach i (*)
 cd $i
 if -e ${station}_fix_tra.stacov then
 # Cabecera del fichero de salida:
 set X=`grep 'STA X' ${station}_fix_tra.stacov | awk '{printf "%15.4f",$5}'`
 set Y=`grep 'STA Y' ${station}_fix_tra.stacov | awk '{printf "%15.4f",$5}'`
 set Z=`grep 'STA Z' ${station}_fix_tra.stacov | awk '{printf "%15.4f",$5}'`
 echo $year $i $ii $X $Y $Z | awk '{printf "%9.4f %15.4f %15.4f
%15.4f\n",$1+($2-0.5)/$3,$4,$5,$6}' >> ../${sta}_xyz.dat
 endif
 cd ..
end
```

Bibliografía

- Berrocoso, M. (2004). *El Sistema de Posicionamiento Global*. Cádiz. Universidad, Servicio de Publicaciones.
 - Bertiger, Willy; Desai, Shailen D.; Haines, Bruce; Harvey, Nate; Moore, Angelyn W.; Ow en, Susan; Weiss, Jan P. (2010). Single receiver phase ambiguity resolution with GPS data. Journal of Geodesy, Volume 84, Issue 5, pp.327-337.
- Changsheng, C. (2009). *Precise Point Positioning Using Dual-Frequency GPS and GLONASS Measurements*. Departament of Geomatics Engeneering. Calgary.
- Esteban Pinillos, H. *Optimización de la transferencia de tiempo y frecuencia mediante el empleo de receptores GNSS geodésicos*. Director: Francisco Javier Galindo Mendoza, Jorge Gárate Pasquín. Universidad de Cádiz, 2012.
- García Silva, C. (2012). Análisis de series temporales GPS y su aplicación al estudio de GPS dual de Ceuta. San Fernando, Cádiz. Real Observatorio de la Armada.
- Gregorius, T. (1996). GIPSY-OASIS How It Works. Newcastle. Ed University of Newcastle.
- Heiskanen, W. Moritz, Helmut. (1967). Physical Geodesy. California. W. H. Freeman.
- Hofmann-Wellenhof, B. Lichtenegger, H. Collins, J. (1997). *GPS: Theory and Practice.* Michigan. Springer-Verlag.

http://igscb.jpl.nasa.gov/

https://gipsy-oasis.jpl.nasa.gov/

http://holt.oso.chalmers.se/loading/

- Kouba, J. (2009). A guide to using international GNSS service (IGS) Products. Central Bureau.
- Leick, A. (1990). GPS Satellite Surveying. Orono, Maine. Wiley-Interscience.
- Perez Peña, A. Modelización de las deformaciones corticales en el Sur de España y Norte de África a partir de observaciones de satélites GPS. Director: José Martín Dávila, Jorge Gárate Pasquín. Universidad de Cádiz, 2007.
- Saastamoinen, J. (1972). Atmospheric correction for the troposphere and stratosphere in radio rangin of tatellites. Geophysical Monograph No. 15, Henriksen, pp 247-251.
- Seeber, G. (2003). Satellite Geodesy. New York. Walter de Gruyter.
- Williams, S. (2008). *CATS : GPS coordinate time series analysis software*. Proudman Oceanographic Laboratory, pp. 147-153.
- Xu, G. (2003). GPS: Theory, Algorithms and Applications. Springer.
- Zumberge, J. F.;Heflin, M. B.; Jefferson, D. C.; Watkins, M. M. y Webb, F. H. (1997). *Precise point positioning for the efficient and robust analysis of GPS data from large networks.*J. Geophys. Res., No 102 (B3), pp. 5005-5017.





