

ASISTENTE A LA PLANIFICACIÓN PARA
MISIONES ESPACIALES DE OBSERVACIÓN
TERRESTRE: APLICACIÓN AL SENSOR SAR DEL
PROYECTO SAOCOM

Alumno:

Daniel Teira

Directores:

Ing. Guillermo Benito†

Ing. Leonardo Handszok†

Codirector:

Dr. Julio Jacobo§

†INVAP S. E. - Div. Modelística

Moreno 1089, San Carlos de Bariloche, 8400 Río Negro, Argentina

§Departamento de Computación - FCEyN

Universidad de Buenos Aires

Planta Baja - Pabellón I

Ciudad Universitaria, 1428, Buenos Aires, Argentina

2 de noviembre de 2005

Índice general

1. Introducción	1
1.1. Marco de desarrollo del software GEOSAR	1
1.2. Principio de Funcionamiento del Radar de Apertura Sintética. Introducción al problema temporal.	4
2. Órbita de satélites artificiales	9
2.1. Definición de la órbita	9
2.2. Perturbaciones	12
3. Teoría de perturbaciones generales	14
3.1. El método de Kozai	14
3.2. La función perturbadora	15
3.3. Perturbaciones de primer orden de período corto.	19
3.4. Perturbaciones seculares de primer orden	20
3.5. Perturbaciones de período largo y perturbaciones seculares de segundo orden	21
3.6. Resultados	22
4. Cálculo de la órbita	23
4.1. Propagador de órbita	23
4.1.1. Propagadores SGP	23
4.1.2. El modelo SGP	24
4.1.3. El modelo SGP4	25
4.2. Implementación del propagador	26
4.2.1. Tests de la rutina del propagador	26
5. Modelo de tierra, y sistemas de coordenadas	35
5.1. Sistemas de coordenadas	35
5.1.1. Sistemas de coordenadas cartesianos	37
5.1.2. Sistemas de coordenadas esféricas polares	40
5.1.3. Transformaciones entre sistemas de coordenadas	41

5.2.	Modelo de tierra	44
5.2.1.	La forma de la tierra	45
5.2.2.	Modelo WGS 84	46
5.2.3.	Modelo de terreno	47
5.3.	Proyecciones de mapas	50
5.3.1.	Proyecciones de mapas en GEOSAR	54
6.	El satélite	55
6.1.	La órbita del SAOCOM	55
6.1.1.	Sistema de coordenadas de la órbita	56
6.2.	El Radar de Apertura Sintética (SAR)	58
6.2.1.	Sistema de coordenadas de la antena	58
7.	Descripción del software GEOSAR	60
7.1.	Introducción	60
7.2.	Presentación	60
7.3.	Contexto de uso	61
7.3.1.	Usuarios	61
7.3.2.	Software	62
7.4.	Funcionalidades de la aplicación	63
7.4.1.	Cálculo de ángulos de pasada	64
7.4.2.	Dibujo del <i>swath</i>	66
7.4.3.	Cálculo de rangos	67
7.4.4.	Capas de información	72
7.4.5.	Persistencia del entorno	73
7.5.	Ejemplo de ejecución	73
7.6.	Modo asistente al planificador	77
7.7.	Esquema de interacción	77
8.	Detalles de implementación	81
8.1.	El lenguaje utilizado: IDL 5.6	81
8.1.1.	Descripción	81
8.1.2.	El lenguaje de programación	82
8.1.3.	Manejo de memoria	84
8.1.4.	Motor gráfico	85
8.2.	Implementación de GEOSAR: esquema general	86
8.2.1.	El propagador	86
8.2.2.	La escena	87
8.2.3.	Los objetos gráficos	87
8.2.4.	Los layers	88
8.2.5.	Esquema de graficación	89

8.3. Comunicación con el planificador	90
8.3.1. Esquema de interacción para calcular rangos 92	
8.3.2. Esquema interacción para graficar una pasada	92
8.3.3. Extensibilidad	93
8.4. Representación del tiempo	93
9. Conclusiones	96
A. Función de fuerza	98
B. Gráficos del ejemplo de uso	100

Índice de cuadros

4.1. Ecuaciones del test	27
4.2. Tabla de resultados del test	29
4.3. Tabla de resultados del test	29
5.1. Clasificación de los Sistemas de Coordenadas según el origen. .	36
5.2. Elementos del sistema ECI-METE	37
5.3. Elementos del sistema ECEF	38
5.4. Definición del elipsoide WGS84	47
5.5. Archivos DEM	49
5.6. Error del DEM	50
6.1. Parámetros orbitales del SAOCOM	56
6.2. Parámetros del sistema del satélite	57
8.1. Archivos de comunicación planificador-GEOSAR	90
8.2. Archivos de comunicación GEOSAR-planificador	91

Índice de figuras

1.1. Esquema de implementación	3
1.2. Apuntamiento electrónico del haz. Ganancia del lóbulo Principal y su relación con el Swath.	5
1.3. Ventana temporal	7
1.4. Deriva del Swath	8
1.5. Diagrama de interferencia en recepción	8
2.1. Esquema órbita	11
4.1. TLE del SAOCOM	24
4.2. Error de posición - Día 1 - METE	30
4.3. Error de posición - Día 1 - ECEF	30
4.4. Error de posición - Día 16 - METE	31
4.5. Error de posición - Día 16 - ECEF	31
5.1. Sistema de coordenadas ECI	38
5.2. Sist. de coord. Geográfico y ECEF	41
5.3. Geoide y elipsoide	46
5.4. Mosaicos GTOPO30	48
5.5. Proyección Cilíndrica	52
5.6. Proyección cilíndrica transversa	52
5.7. Proyección pseudo cilíndrica	52
5.8. Proyección cónica	53
5.9. Proyección cónica oblicua	53
5.10. Proyección Ortográfica	53
6.1. Sistema de coordenadas del satélite	57
6.2. Sistema de coordenadas de la antena	59
7.1. Ventana principal	61
7.2. Región poligonal	64
7.3. Cara visible	66
7.4. Cálculo de ángulos	67

7.5. Rangos en el elipsoiderangos	68
7.6. Rangos en el DEM rangos_dem	69
7.7. Casos de <i>intersecar_elipsoide</i>	70
7.8. Esquema de ejecución	74
7.9. Interacción planificador 1	78
7.10. Interacción planificador 2	79
8.1. Jerarquía de objetos gráficos de IDL	86
8.2. Esquema objetos de la interfaz grafica	89
8.3. Módulo comunicación	92
8.4. Esquema para el cálculo de rangos	92
8.5. Esquema para graficar una pasada	93
B.1. Gráfico de ejemplo 1	100
B.2. Gráfico de ejemplo 2	101
B.3. Gráfico de ejemplo 3	101
B.4. Gráfico de ejemplo 4	102
B.5. Gráfico de ejemplo 5	102
B.6. Gráfico de ejemplo 6	103
B.7. Gráfico de ejemplo 7	103
B.8. Gráfico de ejemplo 8	104
B.9. Gráfico de ejemplo 9	104
B.10. Gráfico de ejemplo 10	105
B.11. Gráfico de ejemplo 11	105

Agradecimientos

A lo largo del trabajo, y de todos estos años que llevo en la facultad y en INVAP, he conocido muchísima gente que me ha ayudado mucho y gracias a la cual pude terminar este trabajo y recibirme.

Creo que toda esa ayuda fue muy importante, y por eso incluyo a todos los que me han dado una mano directa o indirectamente.

Primero quiero agradecer a la familia: a mi mamá Graciela, a mis hermanas Mercedes y Laura; a mi tía Sonia, a Noel y a Verónica.

A los amigos que me acompañaron desde que comencé la facultad: Mario, Esteban, Martín, Pablo, Alejo, Javier, Matías, Juan Pablo y Leonardo.

A la gente de Invap: a mis Directores Guillermo y Leonardo, a Gabriela, a Gustavo Fabbri y a Gustavo Wiman. También quiero agradecer a toda la gente del grupo de Modelística que me han tratado muy bien desde el día en que llegué. A Pablo, Jorge, y a Peter por su ayuda en el trabajo.

A mi Director Julio Jacobo, y a los Jurados Juan Santos y Marta Mejail.

Agradezco también, a la familia Roman: a Olga la mamá de Mario y a Julia, su hermana.

A mis amigos barilochenses: Martín y Gabriel.

Y en general, a todos los que me acompañaron de distintas maneras en estos últimos tiempos.

A todos: MUCHISIMAS GRACIAS!!!.

Resumen

El proyecto SAOCOM (Satélite Argentino de Observación y Comunicación) comprende el desarrollo de un satélite artificial para el transporte de un radar de apertura sintética (SAR) cuya utilidad fundamental será el monitoreo y la prevención de catástrofes.

En el presente trabajo se desarrolló un software especializado cuya tarea es graficar objetos tales como el *groundtrack* (la huella del satélite en La Tierra), el *swath* (el área alumbrada por la antena) y regiones de interés. También puede hacer cálculos de posición y velocidad del satélite, de visibilidad de objetos en la superficie terrestre y de rangos (distancia del satélite a un punto en la tierra). Todas estas funcionalidades, son servicios que se prestan al Planificador de Adquisiciones, que es quien lleva a cabo la configuración específica del radar.

En todo el proyecto, las herramientas y componentes que se desarrollen, deben cumplir con altos requerimientos de precisión y confiabilidad. En este trabajo, se integraron elementos tales como un propagador de órbita, un modelo de elevación del terreno y diferentes proyecciones de mapas, en un entorno gráfico de ventanas, que satisface dichos requerimientos.

Estos elementos fueron implementados en IDL 5.6, aprovechando sus características para el cálculo vectorial y la visualización de datos. Se puso especial atención en la implementación y la validación del propagador, que es una pieza fundamental del trabajo.

A lo largo del trabajo presentado se describen cada una de las partes que interactúan: el propagador, el modelo de terreno, los objetos gráficos y el mecanismo de comunicación con el Planificador. Se detalla la implementación y se dan ejemplos de funcionamiento del producto final.

El resultado, es un software que posibilita optimizar el tiempo de planificación en lo que hace a la determinación del área y el momento a adquirir, permitiendo una planificación más rápida y un mejor tiempo de respuesta ante un requerimiento de adquisición.

Capítulo 1

Introducción

1.1. Marco de desarrollo del software GEOSAR

La presencia argentina en el área de la tecnología espacial a tenido un aumento sustancial en los últimos 15 años y podemos decir que hoy en día forma parte de un selecto grupo de países que pueden proveer esta tecnología. La empresa INVAP S.E., una Sociedad del Estado propiedad del Gobierno de Río Negro, ha construido para la Comisión Nacional de Actividades Espaciales (CONAE), los tres primeros Satélites Argentinos para observaciones científicas, denominados SAC-A, SAC-B y SAC-C, como parte de un programa espacial conjunto entre la CONAE y la NASA de los EE.UU.. Además de los ya mencionados, esta en construcción un sistema satelital de mayor complejidad, denominado SAOCOM. Este, provisto de un Radar de Apertura Sintética (SAR), permitirá la observación de otras características del territorio, así como la visión satelital nocturna y a través de la capa de nubes. Los satélites SAOCOM, de los cuales se fabricarán al menos dos ejemplares, tendrán aproximadamente dos toneladas cada uno y serán los satélites de mayor tamaño y capacidad construidos en el hemisferio sur hasta ahora. Formará parte del sistema Italo Argentino de Satélites para la Gestión de Emergencias (SIASGE), integrado por la CONAE y la ASI (Agencia Spaziale Italiana), que será una constelación de nueve satélites, diseñada para tener información certera y actualizada de incendios, inundaciones, erupciones, terremotos, avalanchas y derrumbes; la utilidad fundamental será el monitoreo y la prevención de catástrofes.

El radar del SAOCOM emitirá microondas a alta potencia, con una antena de 2,5 por 10 metros desplegada en el espacio a más de 620Km de la superficie terrestre. El satélite describirá una órbita helio sincrónica con un periodo de aproximadamente 100 minutos y un ciclo de repetición de 16 días.

Como vigía de catástrofes, este sensor podrá inspeccionar situaciones en forma temprana, independiente de la meteorología, tanto de día como de noche. En este contexto, se ve que el crecimiento y el futuro de esta actividad en nuestro país es muy promisorio.

Debido al altísimo valor agregado que tiene la industria espacial y a la complejidad que manejan estos proyectos, surge la necesidad de contar con herramientas de diseño, planificación y análisis, confiables y específicas que se encuentren a la altura de estas misiones. Cuando no existen en el mercado dichas herramientas adaptables al nivel de especificación requerido, o cuando las que existen no son de fácil acceso por el grado de sensibilidad que tiene la industria espacial, se desarrollan herramientas propias a medida para cubrir necesidades específicas que suelen ser vitales para la misión (en general cada agencia espacial desarrolla sus propias herramientas). Así se incluye, por ejemplo, el desarrollo del software de vuelo o software "duro" que comandará el satélite, el de ingesta, que concierne a la recepción de datos y telemetría, y el de planificación y soporte que, hacen al diseño mismo del sensor y a sus modos de operación.

En ese contexto y en el marco de un convenio entre la Empresa INVAP S.E. y la Universidad de Buenos Aires, se desarrolla este trabajo de Tesis Licenciatura para la carrera de Cs. de la Computación.

El trabajo consiste en el desarrollo de este tipo de herramientas de soporte, que al día de la fecha ya han sido incorporadas y son de uso común dentro del proyecto. En términos generales, consiste en la generación de un paquete de software denominado GEOSAR basado en el manejo de información georeferenciada, preparado para asistir al diseño específico de adquisiciones SAR. El sistema GEOSAR cumple dos objetivos fundamentales: ser una herramienta auto contenida que puede ser ejecutada en modo *stand alone*, y prestar servicio a otros módulos de cálculo que han sido desarrollados para análisis y planificación del instrumento SAR, dentro de la empresa. En la figura 1.1, se muestra un esquema de la ubicación del GEOSAR respecto del software con el que interactúa.

GEOSAR, es una aplicación que posee una interfaz gráfica que facilita la edición, diseño y manejo de distintas capas de información (layers) sobre mapas terrestres de alta densidad. Sobre los mapas, por ejemplo, se pueden definir y visualizar en distintos modos de proyección, zonas de interés, ciudades, ríos, etc. y, fundamentalmente, información específica de interés y uso satelital.

Se ha implementado, en el núcleo de cálculo de GEOSAR, un propagador de órbita usando los estándares SGP4 definidos por NORAD. Con él pueden propagarse orbitas de distintos satélites (no necesariamente el SAOCOM) a

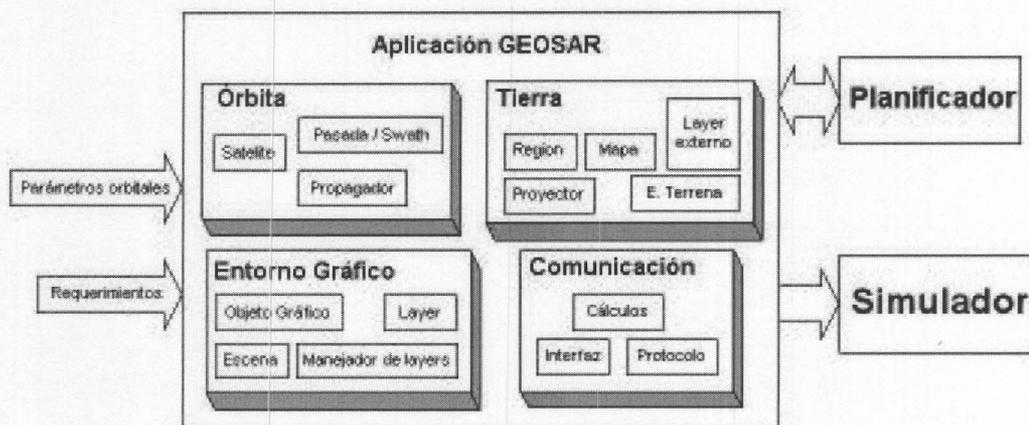


Figura 1.1: Esquema de implementación

partir de datos iniciales - parámetros orbitales - en un formato compatible con esos modelos de propagación, denominado TLE.

Las órbitas son georeferenciadas para ser graficadas en un mapa. Posee un conjunto de herramientas de cálculo concernientes a una aplicación de tipo SAR que ayudan al usuario a planificar una adquisición. Se pueden definir por ejemplo distintos apuntamientos angulares de acuerdo al modo de operación del radar, visualizar la zona iluminada y calcular todos los parámetros temporales necesarios para determinar las ventanas de recepción que serán luego compatibles con los comandos recibidos por el radar. Como se verá más adelante, esto requiere un conocimiento preciso, no sólo de la órbita sino también de las características propias del terreno, para lo cual GEOSAR integra un mapa de altitud (DEM) que rige por el estándar de georeferenciación WGS84.

GEOSAR forma parte de un proyecto mucho más amplio actualmente en desarrollo que incluye entre otros: módulos de planificación, simulación, diseño y evaluación de performance del instrumento SAR, con algunos de los cuales debe interactuar a través de interfaces bien definidas, en donde las funciones del núcleo de cálculo GEOSAR pueden ser invocadas en modo *batch*, a través de llamadas explícitas a su entorno de edición, o, en algunos casos, directamente a un objeto de una de sus clases, insertado en otra aplicación.

Para el diseño, se prefirió una programación orientada a objetos puesto que sus características han permitido, por ejemplo, transferir parte del "kernel" de cálculo GEOSAR al simulador de adquisición con un esfuerzo mínimo.

Los módulos de cálculo de propagación están diseñados con esta concepción, para ser expandidos y reutilizados o, en un futuro, para implementar otros métodos.

El sistema ha sido desarrollado en lenguaje IDL 5.6. Este es un lenguaje matricial basado en C, altamente eficiente en cuanto a la manipulación de grandes volúmenes de información, tratamiento de imágenes, manejo de entidades gráficas y orientación a objetos. Muchos de los programas de uso común en la industria espacial como ENVI han sido desarrollados en este lenguaje.

1.2. Principio de Funcionamiento del Radar de Apertura Sintética. Introducción al problema temporal.

A diferencia de las cámaras comunes, un radar SAR puede “ver” a través de las nubes o de noche, detectar agua bajo cobertura de árboles, medir los cambios de altura de un domo volcánico, e incluso en algunos casos, penetrar el subsuelo para discernir ríos o estructuras subterráneas, o estimar riesgo de inundación o sequía. A diferencia de las cámaras provistas por los satélites ópticos, un radar SAR emite pulsos electromagnéticos. Por ejemplo en el caso del SAOCOM, la longitud de onda de operación ronda los 23 cm o, equivalentemente, los 1300MHz (banda L). Esta banda de frecuencias es muy eficiente para penetrar las distintas capas de la atmósfera minimizando las pérdidas de energía. Los pulsos de radiación de corta duración (aproximadamente 50 μ s) penetran la atmósfera, interactúan con el terreno y una parte de su energía es reflejada y captada nuevamente por el sensor. De acuerdo a las características propias del terreno - su coeficiente de retroreflección- el sensor recibe distinta cantidad de energía. En una aplicación típica se envían varios miles de estos pulsos por segundo mientras que el radar en su recorrido ilumina lateralmente una zona -Swath- de unos 50 o 60 km para uno de sus modos más simples de operación. En modos más especializados, la cobertura puede superar los 300 km. Toda la información recibida por el sensor durante la adquisición (reflejo de los pulsos transmitidos) se reemite a tierra en forma digital cuando el satélite se encuentra dentro del área de cobertura de una estación terrena. Usando algoritmos específicos de descompresión en rango y azimut se puede obtener, a partir de los datos adquiridos (RAW), una imagen de contraste que da muestra de las propiedades del terreno. El uso de esta información y su aplicación es un campo en continua expansión para muchas

disciplinas como la meteorología, la biología, la agricultura, etc.

El componente fundamental del radar es su antena. La misma consiste en un arreglo rectangular de osciladores que emiten coherentemente con una cierta amplitud y fase. La contribución de todos los osciladores produce una ganancia direccional que favorece la emisión de energía en una determinada dirección respecto de las demás. Modificando la fase de los osciladores (Steering electrónico en rango) se logra apuntar el haz sobre la zona de interés sin necesidad de modificar el apuntamiento físico de la antena. Por lo general, la antena posee una inclinación respecto del punto de nadir de aproximadamente 30° (tilt o apuntamiento sin Steering) y puede barrer ángulos de $\pm 15^\circ$ respecto de ese valor.

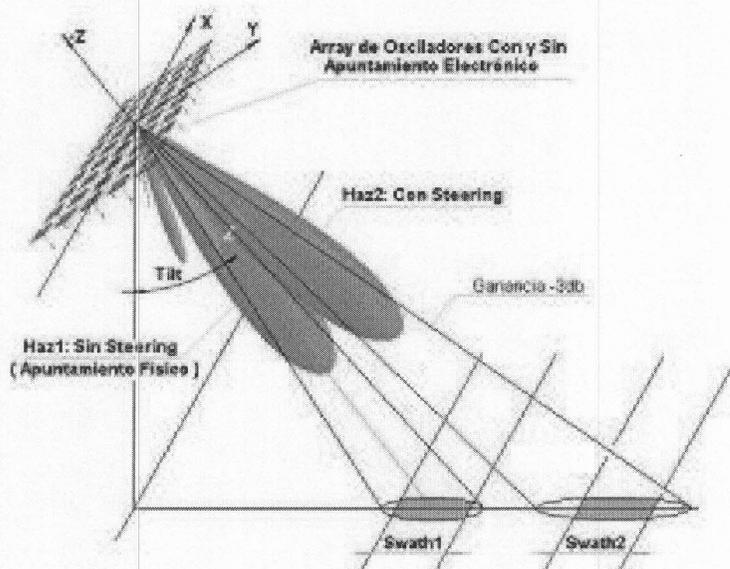


Figura 1.2: Apuntamiento electrónico del haz. Ganancia del lóbulo Principal y su relación con el Swath.

Ganancias típicas del lóbulo principal de una antena SAR se encuentran alrededor de 40 db (potencia emitida en la dirección de máximo apuntamiento respecto de la emisión de una antena omnidireccional con la misma potencia). El ángulo efectivo de una antena SAR es proporcional a la longitud de onda e inversamente proporcional a su ancho. Para una antena de 2.5 m x 10 m como SAOCOM, la apertura angular es de aproximadamente 5° en altitud y 1° en azimut. Se suele estimar la apertura efectiva del lóbulo principal como el valor angular para el cual la ganancia disminuye a la mitad respecto de

su máximo, o equivalentemente, cuando su ganancia decae -3db. La zona de interés o Swath debe estar comprendida dentro de la zona de máxima iluminación mencionada, pero no es el patrón de ganancia lo que determina la porción de terreno adquirida sino la ventana temporal de adquisición. Cuando se emite un pulso, el mismo se propaga esféricamente a la velocidad de la luz. Existe un tiempo de espera a partir de la emisión del pulso en el cual la señal debe recorrer dos veces la distancia que separa al satélite del rango mínimo de interés (Rango cercano del Swath). Esto define el tiempo de inicio para la ventana de recepción. Por otro lado, debe esperarse a que la señal proveniente del rebote en el rango lejano del Swath retorne al sensor antes de apagar el receptor (en realidad a este tiempo de recepción debe sumársele el tiempo de duración del pulso para que el total del pulso transmitido haya sido recibido desde el punto más lejano del swath). Como vimos, el problema del apuntamiento angular es en realidad un problema temporal que depende de la órbita del satélite, y de las características del terreno. Debido a la excentricidad de la órbita y a la forma de la tierra pueden existir variaciones en la altitud del satélite de aproximadamente 30 km. Teniendo en cuenta que una adquisición puede durar varios minutos y que el periodo orbital es de aproximadamente 100 minutos, se puede ver que durante la adquisición pueden existir variaciones en la altitud de kilómetros sin tener en cuenta, además, la rugosidad propia de terreno respecto del nivel del mar. Este hecho conduce, si no se modifican los tiempos de espera y recepción durante la adquisición, a una deriva indeseable de la zona adquirida (la deriva o corrimiento del swath es comparable a la variación de altura orbital en kilómetros). La zona adquirida podría quedar incluso fuera del lóbulo principal de ganancia de la antena.

Existe un ingrediente adicional que complica la determinación temporal de la ventana de adquisición que es el problema de la interferencia de los pulsos de recepción con los pulsos de transmisión y/o con los rebotes provenientes de nadir. En la mayoría de los casos, el sensor posee una única antena SAR que se usa tanto para transmitir, a una determinada frecuencia de repetición de pulsos (PRF), como para recibir. El nivel de señal del pulso recibido es inversamente proporcional a la cuarta potencia del rango recorrido y su nivel se encuentra varios órdenes de magnitud por debajo de la señal emitida. El receptor debe amplificar la señal recibida y no puede emitir pulsos durante ese período. Este hecho conduce a que existan, para una determinada PRF, zonas de apuntamiento prohibidas (distancias prohibidas) para las cuales existe superposición en tiempo entre la llegada de los pulsos y su emisión o, equivalentemente, para un determinado apuntamiento no cualquier PRF esta permitida.

Por lo general se emiten aproximadamente 10 nuevos pulsos en vuelo

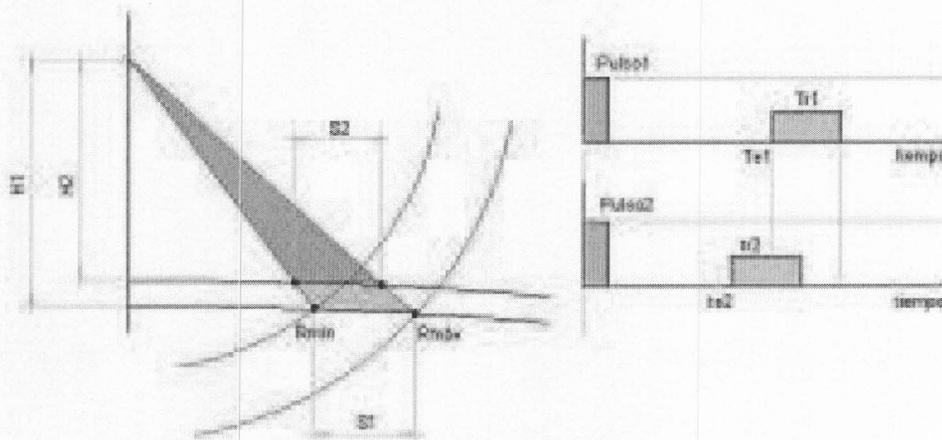


Figura 1.3: Variación de la ventana temporal de Recepción a fin de mantener el apuntamiento constante independientemente de la altura orbital.

antes de recibir el rebote de un determinado pulso emitido. Las variaciones de altura orbital antes mencionadas, sumadas al problema de la interferencia, explican porque es tan necesario contar con un conocimiento preciso de la órbita y de las características topológicas del terreno a fin de estimar correctamente las variables temporales puestas en juego. En este trabajo se ha hecho hincapié en este tema logrando resultados compatibles con los valores de precisión necesarios para la correcta operación del radar.

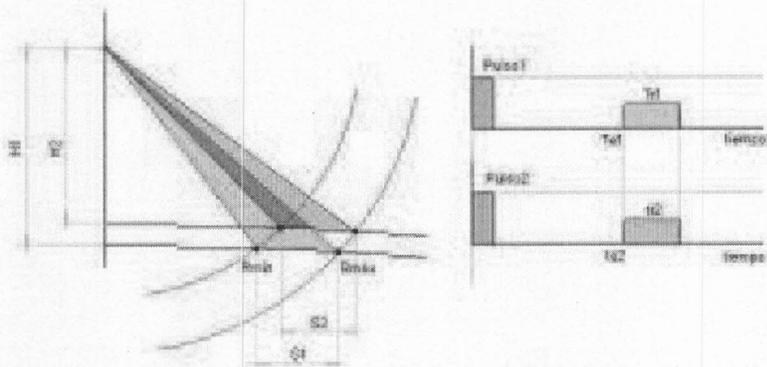


Figura 1.4: Deriva del swath con la altura orbital para una ventana de Recepción cte.

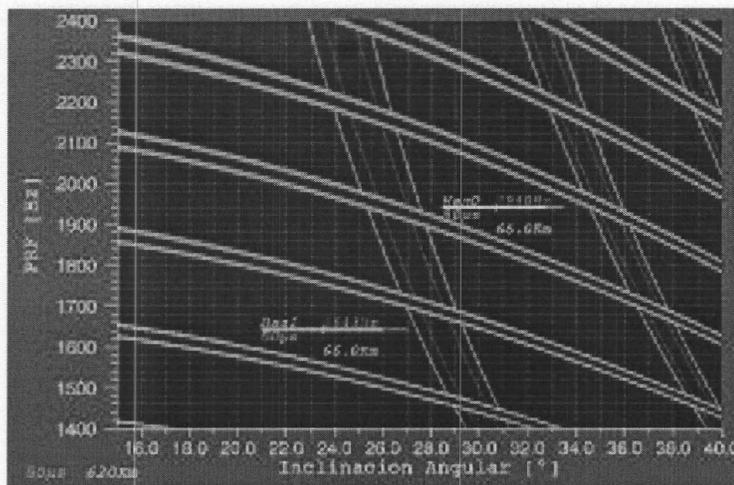


Figura 1.5: Diagrama de Interferencia en Recepción PRF vs Inclinación Angular para una determinada altura orbital. Las franjas azules muestran la interferencia con pulsos transmitidos. Los "haces" (segmentos horizontales) se ubican fuera de la zona de interferencia.

Capítulo 2

Órbita de satélites artificiales

En este capítulo se dan las definiciones principales de los elementos que aparecen en la definición y el tratamiento de órbitas. Se describen las perturbaciones relacionadas con ellas y los métodos de propagación de la órbita (cálculo de la posición y velocidad del satélite).

2.1. Definición de la órbita

Para definir completamente la órbita se debe establecer primero, un sistema de coordenadas, que variará dependiendo del problema. En el capítulo 5 se especifican los utilizados en este trabajo. Una vez elegido este sistema de referencia, se utiliza un conjunto de parámetros que describen cada aspecto y determinan la forma y la orientación de la elipse en el espacio. En este conjunto, también se definen aquellos que marcan la posición del satélite dentro de la órbita.

Así se tienen:

- Para describir la forma de la elipse:
 - Excentricidad (e)
 - Longitud del eje mayor (a) y/o eje menor (b)
- Para ubicar el plano de la órbita:
 - Ascensión derecha del nodo ascendente (Ω)
 - Inclinación (i)

La posición de la elipse dentro del plano:

- Argumento del perigeo (ω)

- Foco de la elipse (f , posicionado en el origen de coordenadas)
- Para ubicar el satélite dentro de la órbita:
 - Anomalía verdadera (ν)
 - Anomalía excentrica (E)

Las anomalías verdadera y excéntrica son ángulos tomados desde el perigeo de la órbita. Ambas se señalan en la figura 2.1. También se utilizan otros elementos, sin significado geométrico, pero con igual importancia. Estos son:

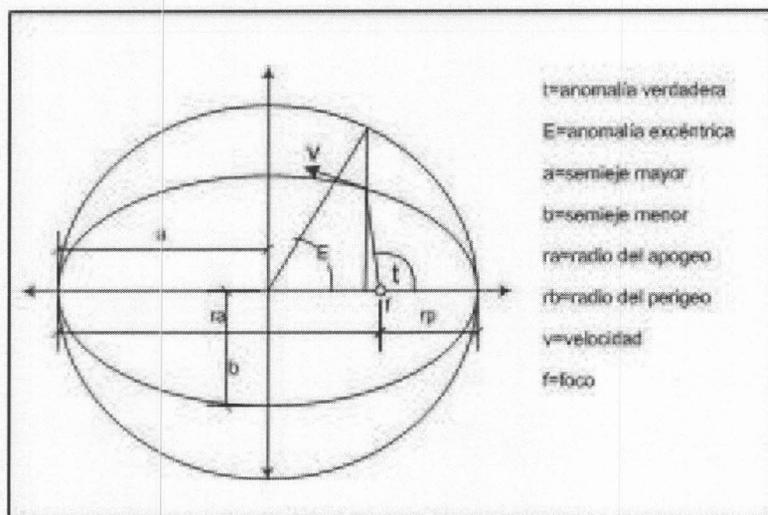
Movimiento medio (n):

se define como $\frac{2\pi}{P}$, donde P es el período de la órbita. Este valor (junto con el semieje mayor) son valores constantes, que dependen solo de la velocidad de un cuerpo a una distancia r del cuerpo principal. El movimiento medio, también se puede obtener de la expresión $n^2 a^3 = \mu$ ($\mu = GM$). Estas expresiones pueden encontrarse en [Zad98]. También está allí la forma de obtención.

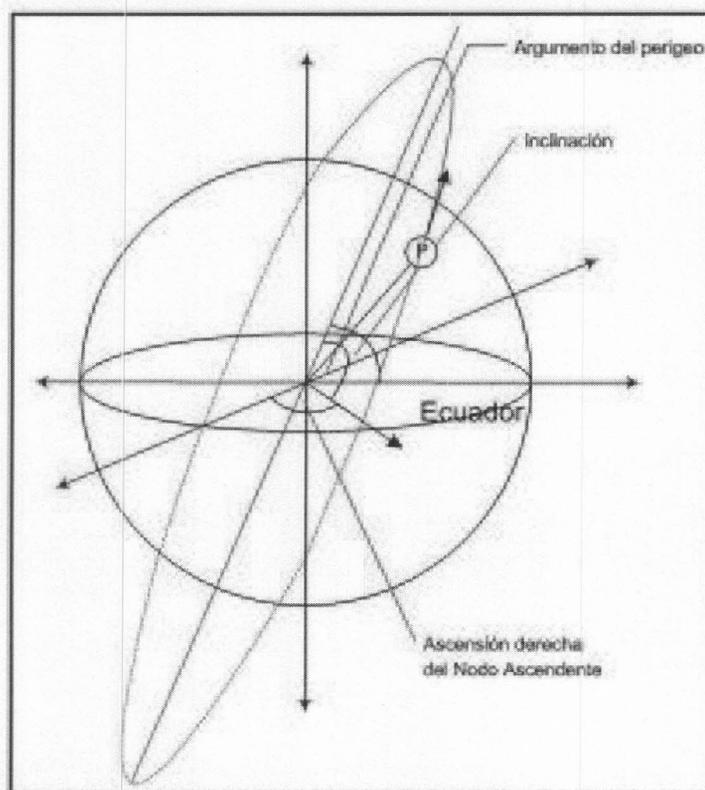
Anomalía media: (M)

Si T es el instante de paso del objeto por el pericentro, t es el instante actual, entonces $n(t - T)$ es el ángulo que describiría el radio vector girando con una velocidad angular media n (movimiento medio). Este valor se denomina anomalía media (*mean anomaly*) y se indica con M .

Todos estos parámetros, para el caso particular de la órbita del SAOCOM, se muestran en la sección 6, en una tabla con el valor correspondiente.



a) Descripción de la elipse



b) Posicionamiento de la elipse en el sistema de coordenadas geocéntrico

Figura 2.1: Esquema de descripción de una órbita

2.2. Perturbaciones

En la situación de una órbita ideal sin que los cuerpos se vean afectados por ninguna condición externa, los elementos orbitales deberían mantenerse siempre constantes (a excepción de las anomalías, que varían a medida que el cuerpo se mueve). Idealmente, si dos cuerpos se mueven bajo la influencia de sus atracciones mutuas, cada uno describe una curva cónica respecto del otro.

En el caso de un satélite en la órbita de la tierra, numerosos factores - *perturbaciones*- afectan las fuerzas intervinientes haciendo que los parámetros orbitales varíen con el tiempo. La influencia de cuerpos externos, o la forma no ideal de ellos mismos, son algunos de los factores que alteran el movimiento de atracción ideal.

Estas perturbaciones mueven y deforman la elipse, modificando los parámetros orbitales, de dos formas posibles, que dan nombre a los dos tipos de perturbaciones:

- **Seculares:** cuando varían en función del tiempo de forma constante.
- **Periódicas:** cuando varían de forma sinusoidal. A su vez, las perturbaciones pueden ser de período corto ó largo, como se verá más adelante.

Para las órbitas terrestres, los factores perturbadores son conocidos e influyen en diferente magnitud y de distinto modo. A continuación, se describen los más importantes:

- **La forma irregular de la tierra:**
Esta es la perturbación más importante de todas en las órbitas alrededor de la Tierra. Como se verá en la sección 5, la tierra no es una esfera perfecta sino que presenta un achatamiento en los polos del orden de los 20Km Esta deformación hace que el campo gravitatorio de la tierra no sea homogéneo y que la fuerza de atracción varíe en cada punto de la órbita.

Estos cambios en la dirección de la fuerza de gravedad, afectan al argumento del perigeo (ω) y el nodo ascendente (Ω). Producen un movimiento de precesión en el plano de la elipse que depende de la inclinación. De esta forma, el movimiento será hacia el oeste para inclinaciones menores que 90° y al este para las mayores. Este efecto -también conocido como

el efecto J_2 se utiliza en las órbitas heliosincrónicas para manejar la incidencia del sol sobre el satélite, con el fin de, por ejemplo, mantener los paneles solares mirando al sol, a lo largo del año (y de la órbita de la tierra alrededor del sol).

El efecto J_2 también se observa en el argumento del perigeo, a través de una rotación de la línea de los nodos como consecuencia de la precesión del plano de elipse. Dicha rotación se da en el sentido del movimiento del satélite para inclinaciones menores que 63.4° o mayores que 116.6° ; y en el sentido opuesto para órbitas con inclinación entre 63.4° y 116.6° . Al igual que en el caso anterior, esta perturbación se realiza de forma secular, es decir que el elemento varía continuamente en el tiempo. Sobre la inclinación, este efecto produce una leve variación periódica en el ángulo, en un sentido a medida que el satélite se acerca al Ecuador, y en el inverso a medida que se aleja.

La deformación del campo gravitatorio no produce cambios seculares en la forma de la elipse y por eso el semieje mayor (a) y la excentricidad (e) permanecen constantes.

- El rozamiento de la atmósfera:
En órbitas de altitud menor a 1000Km, el rozamiento de la atmósfera no es despreciable y produce una fuerza en sentido contrario al movimiento del cuerpo. Dicha fuerza encuentra su magnitud máxima en el perigeo -momento de máxima velocidad y cercanía a la tierra- lo que produce una reducción del eje mayor y la excentricidad, deformando la elipse y llevándola a una forma circular hasta, incluso, llegar a producir la caída del objeto.
- Los efectos por la fuerza de gravedad del sol y la luna:
Estos cuerpos producen ciertas perturbaciones producto de la atracción que ejercen sobre el objeto en órbita. De todos modos, estos efectos son mínimos comparados con el rozamiento de la atmósfera o la deformación de la tierra.
- Los efectos de presión y radiación solar
Una radiación cualquiera ejerce una fuerza en el cuerpo sobre el cual incide. La luz solar genera una aceleración en un satélite, y puede causar perturbaciones a largo plazo.

Capítulo 3

Teoría de perturbaciones generales

Las teorías de perturbaciones generales son, al contrario de las de perturbaciones especiales, modelos semianalíticos de las interacciones de fuerzas que ocurren en el movimiento orbital. En este tipo de teorías se plantea un modelo de fuerzas bajo ciertas suposiciones y se hace un desarrollo analítico. En este desarrollo, se incorporan simplificaciones que, a pesar de resignar precisión en los valores que se obtienen, mantienen un margen de error aceptable respecto de los valores reales y permiten obtener un método de cálculo cerrado (sin integración numérica). Dichas simplificaciones se hacen, por ejemplo, al introducir un valor medio producto de un promedio temporal.

3.1. El método de Kozai

En la presente sección se describe la teoría de perturbaciones generales desarrollada por Y. Kozai en el artículo [Koz59]. Si bien esta teoría no es la base del propagador implementado en el trabajo (SGP4), sí se aplica casi directamente a otro de esa misma serie (SGP) y muestra un ejemplo claro de las teorías de perturbaciones generales, que permite apreciar cuál es su espíritu y como se implementa en una aplicación específica. El modelo SGP4 no utiliza el desarrollo de Kozai. Llega a resultados similares, pero a través de un desarrollo distinto, más complejo.

En las secciones que siguen, se desarrolla este método y se muestra la secuencia de pasos para obtener las funciones perturbadoras de cada elemento orbital.

3.2. La función perturbadora

Toda la teoría que sigue, se hace bajo la suposición de que no hay efectos de rozamiento de la atmósfera, y que el campo de gravedad terrestre tiene simetría axial. El procedimiento comienza determinando la fuerza que va a determinar el movimiento del cuerpo: el potencial gravitatorio. Que para cualquier punto de latitud δ y radio r (distancia geocéntrica) se puede expresar mediante la siguiente sumatoria:

$$U(r, \delta) = \frac{GM}{r} \left(1 - \sum_{i=2}^n J_i \frac{a_e^i}{r} P_i(\cos \delta) \right) \quad (3.1)$$

donde G es la constante gravitatoria y M la masa de la tierra. Los coeficientes J_i se denominan *coeficientes armónicos*. Sus valores se obtuvieron de forma experimental, basados en la observación de satélites. El $P_i(x)$ es el i -ésimo polinomio de Legendre. Esta expresión se obtiene mediante el procedimiento descrito en el apéndice A, para el caso de un sistema de 2 cuerpos.

Esta misma expresión, puede darse de forma equivalente mediante una suma de armónicos, como se muestra a continuación:

$$U = \frac{GM}{r} \left\{ 1 + \frac{A_2}{r^2} \left(\frac{1}{3} - \sin^2 \delta \right) + \frac{A_3}{r^3} \left(\frac{5}{2} \sin^2 \delta - \frac{3}{2} \right) \sin \delta + \frac{A_4}{r^4} \left(\frac{3}{35} + \frac{1}{7} \sin^2 \delta - \frac{1}{4} \sin^2 2\delta \right) + \dots \right\} \quad (3.2)$$

El propósito de este desarrollo, es derivar los términos de perturbaciones seculares de primer orden, y los de perturbaciones periódicas, hasta el segundo orden. Por este motivo, solo se muestran los primeros 4 términos de la sumatoria, y el resto no se toma en cuenta.

Debido a que el satélite se mueve en una elipse, es conveniente expresar su posición en términos de la elipse, a través del siguiente cambio de variables. De esta manera:

$$r = \frac{a(1 - e^2)}{1 + e \cos \nu}$$

$$\sin \delta = \sin i \sin(\nu + \omega) \quad (3.3)$$

donde a es el eje mayor, i la inclinación, e la excentricidad, ω es el argumento del perigeo y ν es la anomalía verdadera (*true anomaly*).

Luego, para obtener la función perturbadora (i.e. cuánto varía la fuerza de atracción respecto de una situación ideal) solo es necesario restar a esta cantidad la fuerza que existiría si la tierra fuese una esfera ($\frac{GM}{r}$).

$$\begin{aligned} R &= U - \frac{GM}{r} \\ &= GM \left[\frac{A_2}{a^2} \left(\frac{a}{r} \right)^4 \left\{ \frac{1}{3} - \frac{1}{2} \sin^2 i \cos 2(\nu + \omega) \right\} \right. \\ &= + \frac{A_3}{a^4} \left(\frac{a}{r} \right)^5 \left\{ \left(\frac{15}{8} \sin^2 i - \frac{3}{2} \right) \sin(\nu + \omega) - \frac{5}{8} \sin^2 i \sin 3(\nu + \omega) \right\} \\ &= + \frac{A_4}{a^5} \left(\frac{a}{r} \right)^5 \left\{ \frac{3}{35} - \frac{3}{7} \sin^2 i + \frac{3}{8} \sin^4 i + \sin^2 i \left(\frac{3}{7} - \frac{1}{2} \sin^2 i \right) \cos 2(\nu + \omega) \right. \\ &\quad \left. + \frac{1}{8} \sin^4 i \cos 4(\nu + \omega) \right\} \end{aligned} \quad (3.4)$$

Por conveniencia para el desarrollo, es necesario transformar la anomalía verdadera, en la anomalía media, que tiene la gran ventaja de ser constante en el tiempo (de hecho esa es la definición de anomalía media). La anomalía media (*mean anomaly*) se relaciona con la anomalía verdadera, en el movimiento no perturbado, a través de la siguiente relación:

$$\frac{dM}{d\nu} = \left(\frac{r}{a} \right)^2 \frac{1}{\sqrt{1-e^2}} \quad (3.5)$$

De 3.5 puede deducirse que $\frac{r}{a}$ y ν son funciones de e y M , que además son periódicas respecto de esta última. A la vez, M es lineal en función del tiempo para el movimiento no perturbado.

Es más, R también es periódica en función de M y ω , y:

- Los términos que no dependen de ω o M se denominan **seculares**
- Los términos que dependen de M y ω son de **período largo**

- Los términos que dependen solo de M son de **período corto**

El siguiente paso es separar los términos según el tipo de perturbaciones que generan. Para eso, primero, se simplificarán los términos con las identidades que siguen. Estas relaciones son promedios temporales, para una sola vuelta de la órbita (se integra la anomalía media entre 0 y 2π).

$$\begin{aligned}\overline{\left(\frac{a}{r}\right)^3} &= \frac{1}{2\pi} \int_0^{2\pi} \left(\frac{a}{r}\right)^3 dM \\ &= (1 - e^2)^{-\frac{3}{2}}\end{aligned}\tag{3.6}$$

$$\begin{aligned}\overline{\left(\frac{a}{r}\right)^5} &= \frac{1}{2\pi} \int_0^{2\pi} \left(\frac{a}{r}\right)^5 dM \\ &= (1 - e^2)^{-\frac{7}{2}} \left(1 + \frac{3}{2}e^2\right)\end{aligned}\tag{3.7}$$

$$\overline{\left(\frac{a}{r}\right)^5 \cos 2v} = \frac{3}{4}e^2(1 - e^2)^{-\frac{7}{2}}\tag{3.8}$$

El resto de los términos donde interviene $\frac{a}{r}$ son nulos:

$$\begin{aligned}\overline{\left(\frac{a}{r}\right)^3 \cos 2v} &= \overline{\left(\frac{a}{r}\right)^3 \sin 2v} = 0 \\ \overline{\left(\frac{a}{r}\right)^4 \sin v} &= \overline{\left(\frac{a}{r}\right)^4 \cos 3v} = \overline{\left(\frac{a}{r}\right)^4 \sin 3v} = 0 \\ \overline{\left(\frac{a}{r}\right)^5 \sin 2v} &= \overline{\left(\frac{a}{r}\right)^5 \cos 4v} = \overline{\left(\frac{a}{r}\right)^5 \sin 4v} = 0\end{aligned}$$

De esta manera, se puede escribir a R como:

$$R = R_1 + R_2 + R_3 + R_4\tag{3.9}$$

donde:

$$\begin{aligned}R_1 &= GM \frac{A_2}{a^3} \left(\frac{1}{3} - \frac{1}{2} \sin^2 i\right) (1 - e^2)^{-\frac{3}{2}} \\ R_2 &= GM \frac{A_4}{a^5} \left(\frac{3}{35} - \frac{3}{7} \sin^2 i + \frac{3}{8} \sin^4 i\right) (1 - e^2)^{-\frac{7}{2}} \left(1 + \frac{3}{2}e^2\right)\end{aligned}$$

$$\begin{aligned}
R_3 &= GM \left\{ \frac{3 A_3}{2 a^4} \sin i \left(\frac{5}{4} \sin i - 1 \right) e (1 - e^2)^{-\frac{5}{2}} \sin \omega \right. \\
&\quad \left. + \frac{A_4}{a^5} \sin^2 i \left(\frac{9}{28} - \frac{3}{8} \sin^2 i \right) e^2 (1 - e^2)^{-\frac{7}{2}} \cos 2\omega \right\} \\
R_4 &= GM \frac{A_2}{a^3} \left[\left(\frac{1}{3} - \frac{1}{2} \sin^2 i \right) \left\{ 1 - \frac{r^3}{a} (1 - e^2)^{-\frac{3}{2}} \right\} + \frac{1}{2} \sin^2 i \cos 2(\nu + \omega) \right]
\end{aligned} \tag{3.10}$$

De estas ecuaciones:

R_1 : Genera perturbaciones seculares de primer orden pues no contiene términos con ω ni con M (o ν).

R_2 : También genera perturbaciones seculares de segundo orden.

R_3 : Genera perturbaciones de período largo. Esto se debe a los términos $\sin(\omega)$ y $\cos(\omega)$

R_4 : Genera perturbaciones de período corto, pues contiene términos dependientes de ω y ν (y por lo tanto M como se vio en 3.5)

El objetivo de la función perturbadora es determinar la variación de los parámetros orbitales a cada momento. Para eso se utilizan las Ecuaciones Planetarias de Lagrange:

$$\begin{aligned}
\frac{da}{dt} &= \frac{2}{na} \frac{\partial R}{\partial M} \\
\frac{de}{dt} &= \frac{1 - e^2}{na^2 e} \frac{\partial R}{\partial M} - \frac{\sqrt{1 - e^2}}{na^2 e} \frac{\partial R}{\partial \omega} \\
\frac{di}{dt} &= \frac{\cos i}{na^2 \sqrt{1 - e^2} \sin i} \frac{\partial R}{\partial \omega} \\
\frac{d\Omega}{dt} &= \frac{1}{na^2 \sqrt{1 - e^2} \sin i} \frac{\partial R}{\partial i} \\
\frac{d\omega}{dt} &= - \frac{\cos i}{na^2 \sqrt{1 - e^2} \sin i} \frac{\partial R}{\partial i} - \frac{\sqrt{1 - e^2}}{na^2 e} \frac{\partial R}{\partial e}
\end{aligned}$$

$$\frac{dM}{dt} = n - \frac{1 - e^2}{na^2e} \frac{\partial R}{\partial e} - \frac{2}{na} \frac{\partial R}{\partial a} \quad (3.11)$$

donde $n^2a^2 = GM$. Un desarrollo de la obtención de estas ecuaciones en 3.11, puede encontrarse en [Zad98]. En el artículo de Y. Kozai (ver [Koz59]) vienen dadas, y a partir de ellas se continúa el análisis.

Recién en este momento comienza la obtención de las perturbaciones seculares y periódicas, reemplazando R_1, R_2, R_3 y R_4 por R en 3.11, según corresponda. Luego se integra para obtener la función de perturbación de cada elemento orbital.

3.3. Perturbaciones de primer orden de período corto.

Para obtener este tipo de perturbaciones, se debe reemplazar R por R_4 en la ecuaciones de 3.11, e integrar considerando a, n, e, i y ω como constantes; salvo en la última expresión, donde n es variable en función del tiempo, luego de obtener la expresión de a (pues $n^2a^2 = GM$).

Antes de comenzar, es necesario pasar de la variable t a ν sustituyendo, dt por $d\nu$ en la expresión en 3.11, utilizando esta otra:

$$dt = \frac{dt}{dM} dM = \frac{1}{n} \left(\frac{r}{a}\right)^2 \frac{1}{\sqrt{1 - e^2}} d\nu$$

Entonces, por ejemplo, para el caso de la inclinación, la función a integrar resulta (luego del cambio de variables, y el reemplazo de R por R_4):

$$di_s = \frac{\cos i}{n^2a^2(1 - e^2) \sin i} \int \left(\frac{r}{a}\right)^2 \frac{\partial R_4}{\partial \omega} d\nu \quad (3.12)$$

La ecuación es integrable analíticamente, y el resultado es:

$$di_s = \frac{1}{4} \frac{A_2}{p} \sin 2i \left\{ \cos 2(\nu + \omega) + e \cos(\nu + 2\omega) + \frac{e}{3} \cos(3\nu + 2\omega) \right\} \quad (3.13)$$

El mismo procedimiento corresponde al resto de los elementos. Para ellos se pueden obtener expresiones análogas, muy complicadas. Por eso,

y resignando precisión para ganar simplicidad, se realiza la sustitución de $\cos jv$ ($j = 1, 2, \dots$) por su valor medio respecto de M :

$$\overline{\cos jv} = \left(\frac{-e}{1 + \sqrt{1 - e^2}} \right)^j (1 + j\sqrt{1 - e^2}) \quad (3.14)$$

Así, el valor medio de di_s (las perturbaciones de período corto de la inclinación) resulta:

$$\overline{di_s} = \frac{-1}{12} \frac{A_2}{p^2} \sin 2i \overline{\cos 2v} \cos 2\omega \quad (3.15)$$

Lo que sigue es el desarrollo de este procedimiento para los demás elementos. Estos resultados, sin los valores medios, pueden encontrarse en [Koz59]. Los valores medios respecto de M para los demás elementos orbitales son:

$$\begin{aligned} \overline{de_s} &= \frac{A_2}{p^2} \frac{1 - e^2}{6e} \overline{\cos 2v} \cos 2\omega \\ \overline{d\omega_s} &= \frac{A_2}{p^2} \left\{ \sin^2 i \left(\frac{1}{8} + \frac{1 - e^2}{6e} \overline{\cos 2v} \right) + \frac{1}{6} \cos^2 i \overline{\cos 2v} \right\} \sin 2\omega \\ \overline{di_s} &= \frac{-1}{12} \frac{A_2}{p^2} \sin 2i \overline{\cos 2v} \cos 2\omega \\ \overline{d\Omega_s} &= \frac{1}{6} \frac{A_2}{p^2} \cos i \overline{\cos 2v} \sin 2\omega \\ \overline{dM_s} &= -\frac{A_2}{p^2} \sqrt{1 - e^2} \sin^2 i \left\{ \frac{1}{8} + \frac{1 + \frac{e^2}{2}}{6e^2} \overline{\cos 2v} \right\} \sin 2\omega \end{aligned} \quad (3.16)$$

3.4. Perturbaciones seculares de primer orden

Las perturbaciones seculares se obtienen reemplazando $R = R_1$, calculando las derivadas parciales, e integrando con e e i constantes. Luego de esto se obtiene:

$$\begin{aligned}
\bar{\omega} &= \omega_0 + \frac{A_2}{p^2} \bar{n} \left(2 - \frac{5}{2} \sin^2 i \right) t \\
\bar{\Omega} &= \Omega_0 - \frac{A_2}{p^2} \bar{n} t \cos i \\
\bar{M} &= M_0 + \bar{n} t \\
\bar{n} &= n_0 - \frac{A_2}{p^2} n_0 \left(\frac{1-3}{2 \sin^2 i} \right) \sqrt{1-e^2}
\end{aligned}
\tag{3.17}$$

donde ω_0 , Ω_0 y M_0 son los valores de la época de referencia de la órbita. El valor n_0 es el movimiento medio (*mean motion*) correspondiente al semieje no perturbado (a_0).

De las ecuaciones de arriba se desprende que los elementos i y e no tienen perturbaciones seculares.

Al modificar el valor de n por \bar{n} , es conveniente utilizar un valor medio para a , distinto de a_0 .

$$\bar{a} = a_0 \left\{ 1 - \frac{A_2}{p^2} \left(1 - \frac{3}{2} \sin^2 i \right) \sqrt{1-e^2} \right\}
\tag{3.18}$$

para que se mantenga la relación $n^2 a^2 = GM$. De esta manera:

$$\bar{n}^2 \bar{a}^2 = GM \left\{ 1 - \frac{A_2}{p^2} \left(1 - \frac{3}{2} \sin^2 i \right) \sqrt{1-e^2} \right\}
\tag{3.19}$$

3.5. Perturbaciones de período largo y perturbaciones seculares de segundo orden

Estas perturbaciones no se incluyen en el modelo SGP. El procedimiento se describe completamente en el artículo de Kozai.

En dicho artículo, también se demuestra la no existencia de perturbaciones de período largo en el semieje mayor.

3.6. Resultados

Los resultados de la teoría de Kozai son (sin incluir las perturbaciones seculares de segundo orden, ni las de período largo):

$$\begin{aligned}
 a &= \bar{a} + da_s & \text{con} & \quad \bar{a} = a_0 \left\{ 1 - \frac{A_2}{p^2} \left(\frac{1-3}{2 \sin^2 i} \right) \sqrt{1-e^2} \right\} \\
 e &= \bar{e} + de_s \\
 i &= \bar{i} + di_s \\
 \omega &= \dot{\omega}_s t + d\omega_s \\
 \Omega &= \dot{\Omega}_s t + d\Omega_s \\
 M &= M_0 + \bar{n}t + dM_s, & \text{con} & \quad \bar{n} = n_0 \left\{ 1 + \frac{A_2}{p^2} \left(1 - \frac{3}{2} \sin^2 i \right) \sqrt{1-e^2} \right\} \\
 n_0^2 a_0^2 &= GM
 \end{aligned}
 \tag{3.20}$$

donde \bar{e} y \bar{i} son valores medios respecto de M y ω , y ω_0, Ω_0 y M_0 son los valores iniciales (los valores orbitales en la fecha de referencia de la órbita).

Capítulo 4

Cálculo de la órbita

Si bien el lanzamiento no se ha producido aún, sí se puede trabajar con la definición de la órbita *nominal*, que es tal cual está hoy. A través de ella se puede conocer la posición y velocidad del satélite, y llevar a cabo todas las funcionalidades planteadas como requerimientos para este trabajo. Es importante decir que un cambio en la órbita no afectará lo hecho, puesto que es un parámetro del programa, que se ingresa a través de los archivos de TLE. Esto se explica con más detalle en el punto siguiente.

4.1. Propagador de órbita

El propagador es la rutina que computa efectivamente la posición y velocidad de un objeto, en función del tiempo y los elementos orbitales. Para esto, se parte de unos valores iniciales de los parámetros orbitales, con una marca de tiempo asociada (la época) y se calcula la perturbación para los instantes de tiempo que le siguen.

Los diferentes modelos de propagación, implementan diferentes modelos de perturbaciones que pueden ir desde uno simple -movimiento de dos cuerpos que se atraen- hasta uno muy complejo que resuelva una ecuación de fuerzas y sea fiel a los datos verdaderos. Claro que, como el propagador está orientado a ser una rutina computacional donde el tiempo de respuesta es un factor crítico, la elección no siempre se inclina por la precisión puesto que ello implica más tiempo de cálculo, lo que no siempre es aceptable.

4.1.1. Propagadores SGP

El propagador implementado es el *Simplified General Perturbation 4 (SGP4)*, que fue desarrollado por el NORAD para el monitoreo y control de los más

de 7500 objetos en órbita. Resultó de un refinamiento de otro propagador -el SGP- también desarrollado por el NORAD pero con un modelo de perturbaciones más restrictivo y simple.

El modelo se detalla en el artículo [FRH88] donde se encuentran las ecuaciones y un listado de código en lenguaje Fortran que las implementa. Allí también se pueden encontrar las distintas versiones de propagadores desarrollados por el NORAD (SGP, SDP4,SGP8 y SDP8).

Parámetros

La rutina tiene como parámetros los elementos orbitales -en un archivo en formato ASCII denominado TLE-, y un instante de tiempo en minutos. Estos datos de TLE son de amplia difusión. Este es uno de los puntos fuertes de este método, y una de las razones de su popularidad, ya que sin ningún esfuerzo adicional, se puede propagar la órbita de todos los satélites de la actualidad. Solo hace falta conseguir el TLE correspondiente.

En la figura 4.1 se presenta un ejemplo de archivo TLE. Una descripción detallada de cada campo puede encontrarse en el sitio web del Dr. T. S. Kelso (<http://www.celestrak.com>). Allí también se puede encontrar gran cantidad de archivos de TLE de objetos en órbita.

Es importante aclarar, que los valores en el TLE no son los reales de la órbita, sino que representan valores medios calculados especialmente para la serie de propagadores SGP. Por esto, no deben ser usados para propagar con otros métodos, ni deben utilizarse alguno de estos propagadores, con datos obtenidos mediante otros procedimientos.

```
SAOCOM-1 16 days 619 km
1 16237U xxxxxx 04306.92944444 +.00000000 +00000+0 +00000+0 6 10
2 16237 97.8892 131.5371 0000100 90.0000 45.0000 14.82154588 00
```

Figura 4.1: TLE del SAOCOM

4.1.2. El modelo SGP

En la implementación del propagador SGP se encuentra, en una versión simplificada, mucha de la teoría de perturbaciones generales explicadas en la sección 3. De lo expuesto en esas secciones, se distinguen en el propagador:

- Las perturbaciones de período corto (ecuaciones en 3.16)

- Las perturbaciones seculares de primer orden (ecuaciones en 3.17)

En algunos casos, la expresión que aparece en el artículo [FRH88] no es exactamente la expresada en esa sección; sino que se trata de una equivalente. Si bien las ecuaciones del propagador no son una implementación directa de las ecuaciones del capítulo 3, gran parte de la teoría de perturbaciones que se ve en él, nace de la comentada allí.

El efecto de rozamiento de la atmósfera

Este modelo incluye las perturbaciones seculares producto del drag de la atmósfera. Se considera de forma muy sencilla a través de unos coeficientes que son parámetros del propagador. Estos coeficientes corresponden a las derivadas primera y segunda del movimiento medio, y se incorporan al calcular el valor perturbado de a , a través de un desarrollo de Taylor del movimiento medio en función del tiempo. Con los coeficientes y el polinomio de Taylor, se pretende dar una idea de como va cambiando esta variable, como función del tiempo.

4.1.3. El modelo SGP4

La decisión de implementar este modelo de propagador, se basó en que es el utilizado por la CONAE. En general, este es el propagador analítico de mayor aceptación dentro de este tipo de aplicaciones.

Este modelo de propagador obtiene resultados similares mediante un procedimiento distinto al del SGP. Distinto en la forma de derivar las perturbaciones seculares y periódicas; además el SGP4 contempla otras -seculares y periódicas de segundo orden-, que no se incluyen en el SGP. Aún así, el método que utiliza es el de derivación de perturbaciones generales, a partir de la función perturbadora detallada en 3.2. La teoría implementada en el SGP4, puede hallarse en el artículo de Brower [Bro59].

Efecto de rozamiento de la atmósfera

Además de llegar a los resultados de forma distinta, en el propagador SGP4 se utiliza otro modelo (más exacto) para el rozamiento. En aerodinámica, la susceptibilidad al rozamiento de un objeto se mide por

un coeficiente balístico

$$B = C_D \frac{A}{m} \quad (4.1)$$

donde, C_D , es el coeficiente de rozamiento, A es la sección cruzada del satélite (cross-section) y m es la masa. Cuanto mayor sea el número, más afectará el rozamiento al objeto. El SGP4 utiliza un parámetro (B^*) que es el coeficiente balístico ajustado al modelo atmosférico del propagador. Este se incorpora en el cálculo de las perturbaciones seculares.

4.2. Implementación del propagador

Existen diversas implementaciones del SGP4, generalmente están escritas en lenguajes como C o Pascal; en este trabajo se implementó una versión en IDL, que debió pasar por un proceso de validación para asegurar la calidad de los resultados obtenidos a partir de ella.

En esta sección, se describirá la implementación y validación del propagador especificado en las ecuaciones del modelo descrito en [FRH88]. En lo que sigue, se detallan los aspectos tenidos en cuenta a lo largo del proceso de implementación de la rutina en este lenguaje en particular. En la sección que sigue se detallan los tests realizados.

4.2.1. Tests de la rutina del propagador

La precisión de los datos calculados con el modelo SGP4 satisface los requerimientos de la aplicación. El objetivo de estos tests, es probar que la IMPLEMENTACIÓN de la rutina es correcta y no, si la precisión del SGP4 alcanza para el uso que se le quiere dar.

Básicamente, se propagó todo el período de repetición de la órbita -16 días- con la implementación en IDL, en los tres sistemas de coordenadas (ECI-METE, ECEF y LLA), y se los comparó con los datos obtenidos de una implementación confiable. De allí se obtuvo el error absoluto y el error relativo de cada punto de la órbita y para cada sistema de coordenadas. En función de ellos, se calcularon: el *Error Absoluto Medio*, el *Error Relativo Medio* y la *Distancia media* entre los puntos generados por cada una de las implementaciones. Todos estos valores, fueron calculados como se detalla en el cuadro 4.1.

Error Absoluto Medio	$\frac{\sum_{t=0}^n x_t - \hat{x}_t }{n}$
Error Relativo Medio	$\frac{\sum_{t=0}^n \frac{ x_t - \hat{x}_t }{\hat{x}_t}}{n}$
Distancia media de los puntos	$\frac{\sum_{t=0}^n \sqrt{(x_t - \hat{x}_t)^2}}{n}$

Cuadro 4.1: Ecuaciones del test

En las ecuaciones, cada x_t es un punto de la órbita en el instante t , calculado con la implementación hecha en IDL. Como se indicó antes, este test se realizó en tres sistemas distintos por lo que x_t es: una tripla de valores (x, y, z) en kilómetros en los dos primeros casos (ECI-METE y ECEF), y una tripla de latitud, longitud y altura (λ, ϕ, h) en el último (LLA).

Asimismo, \hat{x}_t es el valor en el instante t de la órbita, de los datos contra los que se testea.

En base a estos datos se construyeron las tablas que se muestran en cada test, junto con gráficos generados con los datos usados en las sumatorias.

Datos de Test

Los datos utilizados en el test fueron producidos por otra implementación del mismo propagador. Fueron suministrados por la CONAE, que es la entidad especializada en los temas espaciales en el país.

Se considera que dichos datos, son generados correctamente -sin errores de implementación del algoritmo- por lo que se toman de referencia en la comparación con los producidos en este trabajo.

El conjunto de datos para el test se compone de:

- Posición y velocidad del período de revisita completo en coordenadas geográficas (latitud, longitud y altura).
- Posición y velocidad en coordenadas inerciales, sistema ECI-METE para los días uno y dieciséis del período de revisita.
- Posición y velocidad en coordenadas, sistema ECEF para los días uno y dieciséis del período de revisita.

Estos datos fueron generados utilizando el TLE que se muestra en 4.1, también suministrado por la CONAE. En todos los casos, la discretización del tiempo es cada segundo, por lo que un solo día implica la comparación de 86400 datos (1 día = 86400 segundos)

Proceso de TEST

Todo el proceso de testing, se hace en etapas determinadas por el orden de las transformaciones a los distintos sistemas de coordenadas según se describe:

- a) Se empieza con datos en el sistema inercial ECI-METE, que son los salidos directamente de la rutina del propagador sin otro tratamiento adicional que pueda introducir error por operaciones.
- b) Luego siguen los datos en el sistema de coordenadas ECEF. El pasaje solamente implica una rotación de los ejes X e Y , que hace apuntar al primero en dirección del punto vernal.
- c) Por último viene la conversión a latitud, longitud y altura. Esta última es la más compleja y la que más error puede introducir. A primera vista del algoritmo de conversión, que es iterativo, es de esperar que suceda esto.

Corroborar la validez de los datos a lo largo de las transformaciones es, también, una forma de testear los algoritmos de transformación entre los sistemas. Estos, se describen al final de la sección 5.1, donde se encuentra la definición de cada sistema.

Resultados

De los test realizados se generaron una serie de gráficos y tablas que, se detallan a continuación.

- Tablas de resultados para los días 1 y 16: en cada una de las tablas que se muestran, los valores de las filas corresponden a los obtenidos al aplicar las ecuaciones 4.1, en cada una de las coordenadas. También se da el error absoluto máximo de la muestra.

	Sist. Coord.	Err. Abs. Med.	Err. Rel. Med.	Dist. Media (mts)
Día 1	ECI-METE	X: 0.00001728 Y: 0.00001870 Z: 0.00002240	X: -0.00010513 Y: 0.00049990 Z: 0.00139634	0.03766086
	ECEF	X: 0.00001599 Y: 0.00001677 Z: 0.00002240	X: 0.00112691 Y: -0.00005053 Z: 0.00139634	0.03746513
	LLA	LAT:0.00002593 LON:0.00002889 ALT:0.00025148	LAT:-0.01323115 LON:0.01888431 ALT:0.03425697	N/A

Cuadro 4.2: Tabla de resultados del test

	Sist. Coord.	Err. Abs. Med.	Err. Rel. Med.	Dist. Media (mts)
Día 16	ECI-METE	X: 0.00066600 Y: 0.00050864 Z: 0.00074980	X: 0.00241362 Y: 0.01174658 Z: -0.05027074	1.24864478
	ECEF	X: 0.00058539 Y: 0.00057972 Z: 0.00074980	X: 0.14182152 Y: 0.00898394 Z: -0.05027074	1.25967434
	LLA	LAT:0.00058539 LON:0.00057972 ALT:0.00074980	LAT:0.14182152 LON:0.00898394 ALT:-0.05027074	N/A

Cuadro 4.3: Tabla de resultados del test

- Gráficos de error para el Día 1 Los gráficos que se muestran corresponden a la diferencia entre los puntos de la órbita de los datos de test, y los generados por la implementación en IDL para el día 1 del período de revisita. El eje x es tiempo en segundos y el eje y es la diferencia en metros.

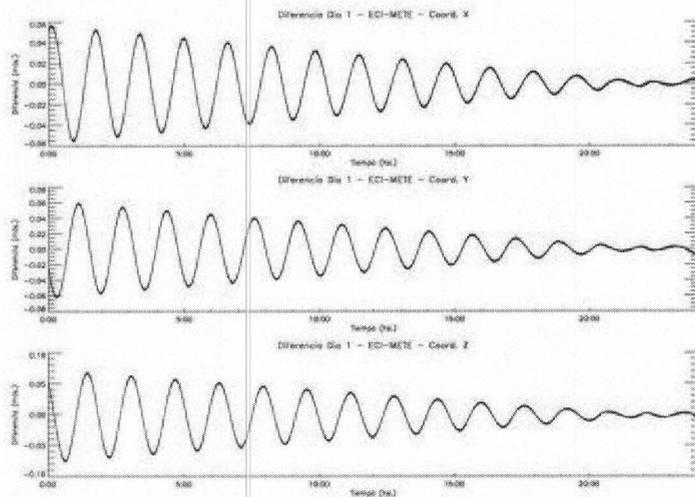


Figura 4.2: Error de posición - Día 1 - METE

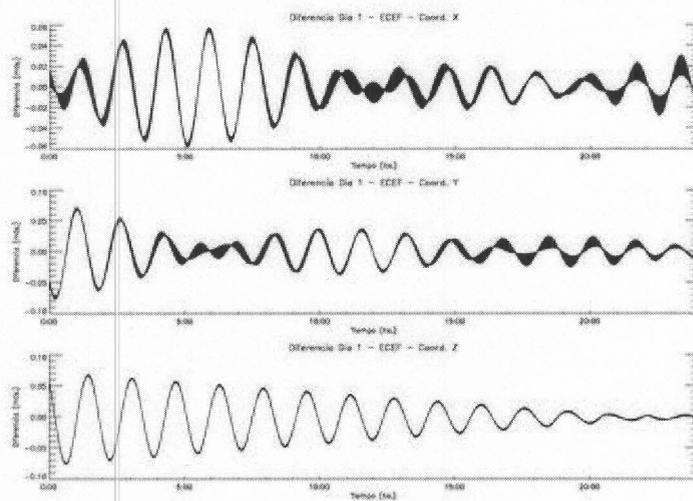


Figura 4.3: Error de posición - Día 1 - ECEF

- Gráficos de error para el Día 16: Los gráficos que se muestran corresponden a la diferencia entre los puntos de la órbita de los datos de test, y los generados por la implementación en IDL, para el día 16 del período de revisita. El eje x es tiempo en segundos y el eje y es la diferencia en metros.

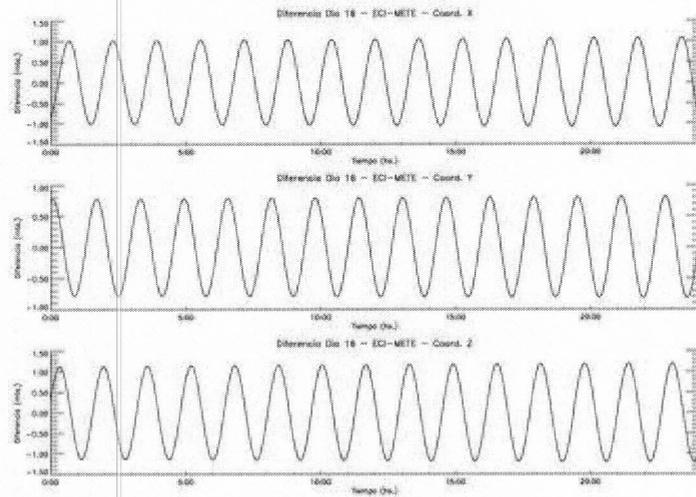


Figura 4.4: Error de posición - Día 16 - METE

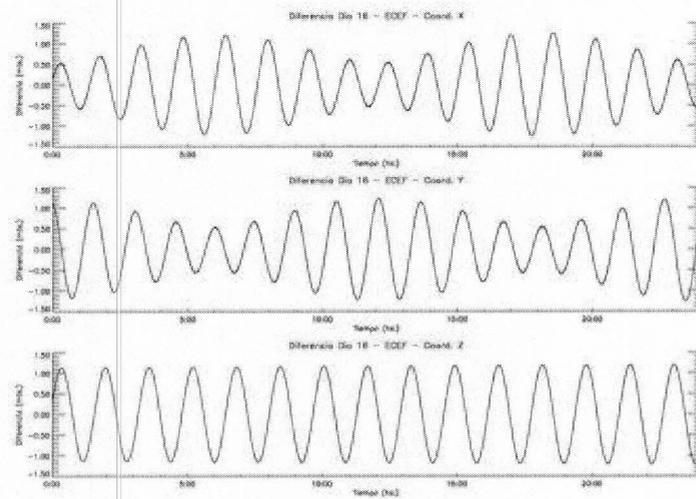


Figura 4.5: Error de posición - Día 16 - ECEF

Criterio de aceptación

Teniendo en cuenta las diferencias que puede haber entre la implementación del propagador que generó los datos de test, y la desarrollada en este traba-

jo - diferencias que en la operación con números de punto flotante, pueden ser críticas - se determinó que el error máximo esperado, (considerando los errores de redondeo y de precisión en los cálculos) debe ser del orden de los 10 metros.

Conclusiones

Las conclusiones que se muestran, se obtuvieron de todos los tests y de la evolución del propagador a lo largo del tiempo. Las más importantes, se describen en los puntos que siguen:

- Del análisis de las tablas 4.2 y 4.3 , puede decirse que la primer transformación (ECI a ECEF) no introduce ningún error significativo. La segunda transformación no es comparable pues el error de latitud y longitud está en grados. Sí se puede comparar la altura, que tampoco sufre grandes modificaciones. De todos modos, debido a la naturaleza de los algoritmos de transformación (en su mayoría rotaciones de los ejes de coordenadas) no deberían haber grandes cambios en la altura. Aún cuando el error aumentó con el paso del tiempo, en los tres casos, está dentro de los valores tolerados.
- La precisión que se use para medir el tiempo es crítica pues el algoritmo SGP4 hace todos sus cálculos en función de una diferencia de tiempos ($t - t_0$).
Además, la resta es una operación que aumenta enormemente la magnitud del error cuando se hace entre números parecidos; y este es el caso del propagador, puesto que al utilizar el TLE más actual -el de época más cercana a la fecha a propagar-, la diferencia ($t - t_0$), hace que se pierdan casi todas las cifras significativas.
En relación con la implementación en IDL, este aspecto se manifiesta en la representación de las fechas que se utiliza en el programa. Las rutinas de conversión a juliano implementadas en el lenguaje, tienen una precisión de 5 decimales significativos. Entonces, como cualquiera de estas fechas del año 2000, tiene siete cifras enteras, luego de la diferencia ($t - t_0$), un número de doble precisión, pierde casi la mitad de su capacidad.
En las rutinas de test, se ve como la fracción de los segundos tiene un error no despreciable (el satélite viaja a $7500 \frac{m}{seg}$), y que al mejorar la precisión, la diferencia es notable.

La solución a este problema, es implementar el manejo de fechas con un origen más cercano (por ej. 1/1/1950), lo que deja más dígitos libres para expresar diferencias mucho más finas que un segundo. Eso fue lo que se hizo en este caso. También existen otras representaciones equivalentes. Un ejemplo son los julianos modificados, que se obtienen restando 2400000.5 a los julianos tradicionales.

- En el gráfico del Día 1 en coordenadas ECI-METE, puede verse que el error disminuye hacia el final del día. Este punto está relacionado con el primero los items de esta sección, puesto que si se analiza el vector de error, se puede ver que el mínimo está en torno de las 22:18hs, que es exactamente la época del TLE. El número de la época es 04306.92944444, donde:
 - 04 es al año (2004)
 - 306 es el día (1/11/2004)
 - 0.92944444 es la fracción del día (22:18:23.999)

En este punto es evidente que el propagador es mucho mejor a medida que se acerca a la época (esta disminución del error no se ve en ningún otro día), y que la precisión de las fechas es muy importante (en los ejemplos con julianos tradicionales esta mejora no se puede percibir). Asimismo, a medida que el tiempo se aleja de la época, el error aumenta, aunque se mantiene dentro de los valores aceptados.

- Una última consideración sobre la precisión de las variables en general. La rutina trabaja con valores de distancia normalizados. Se dividen por el valor del radio terrestre (6378.135 Km) y al terminar el cálculo, se debe volver a multiplicar por dicho número. Si además se considera que el error se analiza en metros, entonces, en total, cada valor que se obtenga se multiplicará, junto con el error, por un factor de 10^6 . Esto nos muestra que cambios mínimos en un número (en el 6to decimal), pueden generar variaciones del orden de los metros.

Como se describe en la sección 8.1, IDL es un lenguaje sin tipado explícito. Lo infiere del contexto y la expresión, y asigna el más chico posible.

Esto que a veces es muy cómodo, tiene efectos indeseables: al no declararse

explícitamente el tipo de las variables, se puede estar perdiendo precisión en el cálculo sin que eso sea advertido.

Capítulo 5

Modelo de tierra, y sistemas de coordenadas

5.1. Sistemas de coordenadas

Los sistemas de coordenadas surgen ante la necesidad de un sistema de referencia para las mediciones en el espacio (en el sentido geométrico). Es necesario poder ubicarse dentro de un sistema que puede variar según los requerimientos de entrada o salida de los datos. Por ejemplo, mediciones hechas desde una posición en la tierra, referidas a un sistema de coordenadas topocéntrico, deberán ser transformadas a uno heliocéntrico, si se desea utilizar esos datos para obtener efemérides de algún planeta o satélite. En este caso se ve, no solo la existencia de dos sistemas distintos, sino también la necesidad de transformar los datos de uno a otro.

En esta sección se describen los sistemas de coordenadas utilizados en el trabajo. Las conversiones que se describen más adelante, se encuentran implementadas en rutinas escritas en IDL. No se muestra el código, solo el procedimiento para llevarlas a cabo. Para una explicación detallada sobre este asunto, consultar [NAS89] o [Sei92].

Definición de un sistema de coordenadas

Para definir completamente un sistema de coordenadas es necesario determinar una serie de elementos:

1. Origen de coordenadas.
2. Plano de referencia.

3. Dirección principal.

En general, estos parámetros varían en un conjunto de uso común que, además, dan nombre al sistema de coordenadas. Así, para el origen de coordenadas, se tiene:

Origen	Designación
Observador	Topocéntrico
Centro de la tierra	Geocéntrico
Centro de masa	Baricéntrico

Cuadro 5.1: Clasificación de los Sistemas de Coordenadas según el origen.

Los planos de referencia que se usan son:

Horizonte el plano tangente a un punto en el elipsoide del modelo de tierra elegido. El norte es la dirección de la normal de dicho plano, en la dirección que se aleja del elipsoide.

Ecuador para sistemas geocéntricos, el plano normal al eje de rotación de la tierra. El norte está en dirección del vector de velocidad angular de rotación, también llamado *polo celestial*. Con respecto a otro cuerpo (e.g. la luna), se define de manera correspondiente.

Plano de la órbita definido para el movimiento de dos cuerpos (*two-body motion*). El norte está en dirección del momento angular.

Eclíptica es el plano de la órbita de la tierra alrededor del sol. El norte está definido en dirección del momento angular del sistema.

La dirección principal se obtiene al intersecar otro plano con el de referencia. Este nuevo plano puede ser meridional (contiene al eje de rotación del cuerpo principal), ecuatorial, u otro plano orbital.

Las direcciones de uso común son:

Greenwich (meridiano principal) El meridiano de Greenwich. Pasa por el primer Royal Observatory en Greenwich, Inglaterra.

Equinoccio vernal (punto vernal) se define como la intersección de la eclíptica y el plano del ecuador terrestre con dirección positiva desde la tierra hacia el sol en el momento en que el sol pasa por el ecuador de sur a norte (punto vernal, aproximadamente el 21 de marzo).

Nodo ascendente es la intersección del plano orbital y el plano de referencia, positiva en la dirección de la intersección del nodo con el plano de referencia, cuando pasa de sur a norte. El equinoccio vernal es un caso de Nodo ascendente.

5.1.1. Sistemas de coordenadas cartesianos

Como su nombre lo indica, la característica fundamental de estos sistemas de coordenadas es que los puntos se expresan como ternas de valores que representan distancias al origen de coordenadas.

Hay dos clases que se detallan en los siguientes cuadros:

1. Earth Centered - Earth Inertial (ECI)

Los parámetros que lo forman son:

Origen	Centro de la tierra
Plano de referencia	Ecuador
Dirección principal	Equinoccio Vernal de la época

Cuadro 5.2: Elementos del sistema ECI-METE

Los ejes se definen como:

- El eje \hat{x} : la dirección principal.
- El eje \hat{y} : normal al \hat{x} y \hat{z} para formar un sistema dextrógiro.
- El eje \hat{z} : normal al plano principal, en la dirección del vector del momento angular principal.

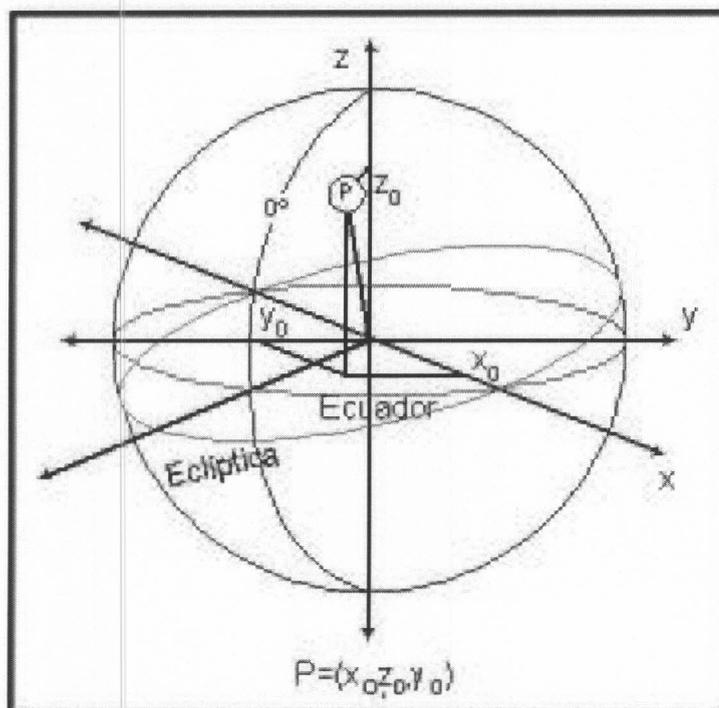


Figura 5.1: Sistema de coordenadas ECI

2. Earth Centered - Earth Fixed (ECEF)

Los parámetros que lo forman son:

Origen	Centro de la tierra
Plano de referencia	Ecuador
Dirección principal	Intersección del ecuador y el meridiano de Greenwich

Cuadro 5.3: Elementos del sistema ECEF

Los ejes se definen como:

- El eje \hat{x} : la dirección principal.
- El eje \hat{y} : normal al \hat{x} y \hat{z} para formar un sistema dextrógiro.

- El eje \hat{z} : normal al plano principal, en la dirección del polo norte celestial.

Más adelante, en la figura 5.2, se muestra un esquema de este sistema de coordenadas, junto con el geográfico.

Sistemas ECI y ECEF

La diferencia fundamental entre estos sistemas, es que, el ECEF, al estar sujeto a un meridiano, rota con la tierra. Es decir que una posición (x, y, z) en la superficie de la tierra siempre tendrá las mismas coordenadas dentro de este sistema.

Esto no sucede con el sistema ECI que no se mueve con la tierra y los puntos de la superficie cambian constantemente sus coordenadas.

A su vez, el sistema ECI cambia debido a una precesión de los equinoccios, producto de la precesión del eje de rotación terrestre. Esto hace variar el ángulo entre la eclíptica y el ecuador -y por lo tanto la dirección principal- haciendo que los ejes se muevan alrededor de 50 arcsec por año; y que el conjunto de efemérides que caracterizan un punto en el sistema ECI, deban venir acompañadas por un tiempo de referencia (epoch) que defina el momento de la medición.

Existen algunos de uso común que varían levemente, estos son:

- ECI-J2000: la dirección corresponde a la de las 0Hs UT del 1 de enero de 2000
- ECI-B1950: la dirección corresponde a la de las 22:9:7.2 del 1 de enero de 1950
- ECI-METE (*True Equator Mean Equinox*): la dirección corresponde al ecuador verdadero y al equinoccio medio (sin considerar la nutación del polo terrestre) del momento de la medición
- ECI-TETE (*True Equator True Equinox*): la dirección corresponde al ecuador y equinoccio verdaderos (considerando la nutación del polo terrestre) del momento de la medición

Las conversiones de uno a otro se obtienen aplicando rotaciones en \hat{y} o \hat{z} a cada sistema de coordenadas según corresponda. Si se quiere cambiar el momento de referencia, se debe rotar el sistema según el ángulo de precesión y nutación.

A lo largo del trabajo, se utilizan los términos ECI y ECI-METE indistintamente, y se considera que hacen referencia al mismo sistema de coordenadas.

5.1.2. Sistemas de coordenadas esféricas polares

A diferencia de los descriptos antes, este sistema no es cartesiano. Sus coordenadas son ángulos en el plano de referencia (x, y) ; y en el plano que forman la dirección principal y el eje de rotación terrestre (x, z) .

Los componentes (origen, plano de referencia y dirección principal), y los ejes, son los mismos que en el sistema ECEF del cuadro 5.3.

Existen dos sistemas distintos, que varían en la forma de definir las coordenadas. Estos son:

- Geocéntrico: Los nombres de las componentes de una coordenada son latitud, longitud y radio, y se notan ϕ' , λ y r respectivamente. Cada una es:

r : la distancia radial al origen del sistema de coordenadas.

ϕ' : el ángulo de longitud positivo desde el meridiano principal

λ : el ángulo de latitud positivo hacia el norte desde el ecuador

- Geodésico: Los nombres de las componentes de una coordenada son latitud, longitud y altura, y se notan ϕ , λ y h respectivamente. Cada una es:

h : la altura hasta el punto, sobre el elipsoide y medida en el sentido normal a la horizontal local.

ϕ : el ángulo de longitud positivo desde el meridiano principal

λ : el ángulo de latitud positivo hacia el norte desde el ecuador hasta el vector normal a la superficie del elipsoide, en el punto medido.

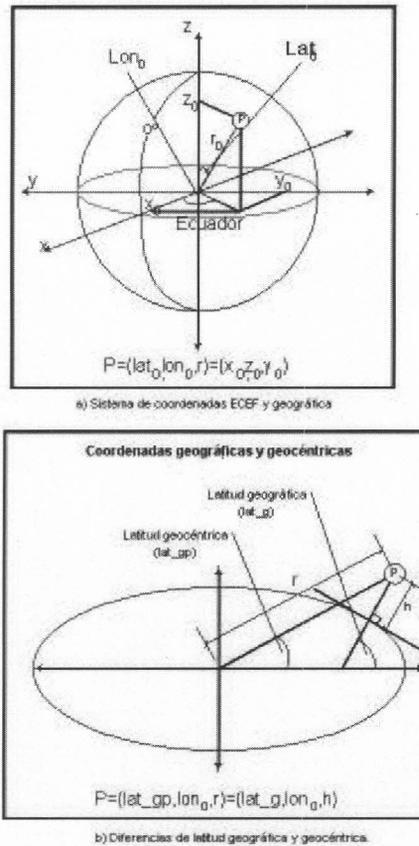


Figura 5.2: Sistema de coordenadas geográficas y ECEF

5.1.3. Transformaciones entre sistemas de coordenadas

Matrices de rotación

En general las transformaciones entre sistemas consistirán en rotaciones de uno o más ejes. Por ejemplo, si un vector tiene en un sistema de coordenadas $S(x, y, z)$, coordenadas x, y y z , sus componentes en otro sistema obtenido por la rotación de un ángulo α en el sentido de las agujas del reloj, alrededor del eje x serán

$$\begin{aligned} x' &= x \\ y' &= y \cos \alpha - z \sin \alpha \\ z' &= y \sin \alpha + z \cos \alpha \end{aligned}$$

(5.1)

Que se puede escribir de forma matricial como

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \cos \alpha & -\text{sen } \alpha \\ 1 & \text{sen } \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

De esta forma se generan las coordenadas en un sistema con una rotación del eje x .

Análogamente, se generan tres matrices, para las rotaciones en cada uno de los ejes x, y y z .

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & \cos \alpha & -\text{sen } \alpha \\ 1 & \text{sen } \alpha & \cos \alpha \end{pmatrix}$$

$$\mathbf{R}_y(\alpha) = \begin{pmatrix} \cos \alpha & 0 & \text{sen } \alpha \\ 0 & 1 & 0 \\ -\text{sen } \alpha & 0 & \cos \alpha \end{pmatrix}$$

$$\mathbf{R}_z(\alpha) = \begin{pmatrix} \cos \alpha & -\text{sen } \alpha & 0 \\ \text{sen } \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Para conversiones que impliquen la rotación de más de un eje, solo se deben multiplicar sucesivamente las matrices que correspondan. Por ejemplo, si se quiere rotar el vector (x, y, z) , primero sobre el eje x y luego sobre el z , se deberá multiplicar:

$$x' = \mathbf{R}_z \mathbf{R}_x x \quad (5.2)$$

Estas matrices tienen la propiedad de ser ortogonales, es decir que se cumple:

$$\mathbf{R} \mathbf{R}^t = \mathbf{I} \quad (5.3)$$

De esta propiedad se desprende que la inversa de una de estas matrices es su traspuesta y se obtiene el método para invertir la conversión y recuperar las coordenadas en su sistema original. Por ejemplo, si se aplica una transformación

$$x' = \mathbf{R}x \quad (5.4)$$

se puede recuperar x a partir de y multiplicando dicho vector por la matriz traspuesta de \mathbf{R} de la siguiente manera

$$x = \mathbf{R}^t x' \quad (5.5)$$

Transformación ECI-METE a ECEF

La transformación entre estos sistemas se lleva a cabo mediante una rotación del plano xy alrededor del eje z . El ángulo que se utiliza para dicha rotación, corresponde al que forma el meridiano de Greenwich con el punto vernal, denominado *Greenwich Mean Sidereal Time (GMST)*. Este ángulo (a_{GM}) se calcula mediante la siguiente expresión:

$$a_{GM} = t \omega + [6^h 41^m 50^s,54841 + 8640184^s,812866 T_u + 0^s,093104 T_u^2 - 6^s,2(10^{-6} T_u^3)] \left(\frac{2\pi}{86400} \right) \quad (5.6)$$

donde:

ω : es la velocidad de rotación de la tierra en radianes por segundo,

t : es el tiempo en segundos desde la 0^h UT1 del día para el que se calculen las efemérides,

T_u : es el número de siglos Julianos transcurridos desde la época J2000.0 (2451545.0).

Una vez calculado el ángulo, solo hay que rotar el sistema alrededor del eje x para lo que se utilizará la matriz $R_z(a_{GM})$, así:

$$\begin{aligned} \vec{x}_{ecef} &= R_z(a_{GM}) \vec{x}_{eci} \\ \vec{x}_{eci} &= R_z^t(a_{GM}) \vec{x}_{ecef}. \end{aligned}$$

Transformación ECEF a Geográfico

La transformación de coordenadas al sistema Geográfico se produce en dos pasos. El primero, para llevar las coordenadas cartesianas a los ángulos de latitud y longitud, para una tierra de forma esférica. El segundo, considerando una tierra de forma elipsoidal.

Los parámetros son (x_b, y_b, z_b) , a partir de las cuales se obtienen (λ, ϕ, r) , (longitud, latitud y el radio respectivamente). La longitud puede obtenerse geoméricamente a partir de x_b y y_b calculando:

$$\lambda = \tan^{-1} \left(\frac{y_b}{x_b} \right) \quad 0 \leq \lambda \leq 2\pi. \quad (5.7)$$

Para calcular la latitud y la altura, en cambio, es necesario utilizar un procedimiento iterativo que satisface los requerimientos de calidad necesarios. Utilizando

$$z_i = -e^2 z_b \quad (5.8)$$

como valor inicial para z_i , se iteran las ecuaciones

$$z_{ib} = z_b - z_i$$

$$N + h = \sqrt{x_b^2 + y_b^2 + z_{ib}^2} \quad (5.9)$$

$$\sin \phi = \frac{z_{ib}}{N + h} \quad (5.10)$$

$$N = \frac{R_e}{\sqrt{1 - e^2 \sin^2 \phi}} \quad (5.11)$$

$$z_i = -N e^2 \sin \phi. \quad (5.12)$$

Luego de la convergencia del algoritmo, la latitud y la altura (ϕ y h) se obtienen a partir de las ecuaciones 5.10 y 5.9.

Transformación Geográfico a ECEF

Esta transformación permite el pasaje de coordenadas de latitud (λ), longitud (ϕ) y altura (h) a coordenadas en (x, y, z) . El procedimiento es directo, mediante tres ecuaciones:

$$\begin{aligned} x &= (h + N) \cos \lambda \cos \phi \\ y &= (h + N) \cos \lambda \sin \phi \\ z &= (h + (1 - e^2)N) \sin \lambda, \end{aligned} \quad (5.13)$$

donde N es el definido en la ecuación 5.11.

5.2. Modelo de tierra

La tierra puede concebirse en sucesivas aproximaciones que van desde la más simple y alejada de la realidad -una esfera- hasta la más compleja y parecida a la tierra real -geoide o un modelo de elevación. Así, a medida que el modelo se hace más complejo, aparecen aspectos que deben considerarse pues modifican el campo gravitatorio y su condición deja de ser la ideal, es

decir que el vector de gravedad varía en cada punto.

Estas mismas aproximaciones deben hacerse con el movimiento de rotación y traslación de la tierra, alrededor de su eje y del sol respectivamente, pues estos movimientos y deformaciones generan perturbaciones que también afectan a los objetos en órbita, que es lo que interesa en este trabajo.

5.2.1. La forma de la tierra

La forma de la tierra interesa especialmente puesto que es -junto con el rozamiento de la atmósfera- la principal generadora de perturbaciones en la trayectoria de los objetos en órbita. En los párrafos que siguen se describen las distintas aproximaciones que se han hecho hasta llegar al cuerpo que mejor se adapta a la forma del planeta -el geoide- y a la representación de la altura verdadera en cualquier punto de la tierra.

El elipsoide

Inicialmente se creía que la tierra era una esfera perfecta. Luego se comprobó que esto no era así y que presentaba un achatamiento en los polos. Que el diámetro del ecuador era levemente mayor que el diámetro en los polos. Este achatamiento -del orden de los 20km- se genera producto del movimiento de rotación de la tierra que, para mantener el equilibrio, debe llevar los fluidos al ecuador.

Si bien parece pequeña, esta deformación no es despreciable pues afecta de manera considerable al potencial gravitatorio de la tierra. Se introduce entonces, una nueva aproximación, y con ella otra figura: un elipsoide. Es decir, una elipse girando en torno al eje menor (el eje de rotación terrestre). Para expresar esta diferencia se utiliza un coeficiente $-f-$ que mide la relación entre los ejes mayor y menor (a y b respectivamente), y se calcula como $\frac{(a-b)}{a}$.

El geoide

El vector de gravedad en cada punto de la tierra se puede medir con mucha precisión. Luego, este resultado se puede comparar con el vector normal a la superficie del elipsoide en ese punto, obtenido de forma analítica. Con este tipo de comparaciones, se determinó que la figura del elipsoide tampoco reflejaba la verdadera forma del campo de gravedad terrestre. Así es como se introduce un elemento nuevo: el *geoide*.

Este cuerpo nuevo, puede pensarse gráficamente como la tierra, sin relieve y cubierta completamente por el océano. Es decir, el estado natural de los fluidos sin perturbaciones por los continentes. Esto representa el potencial gravitatorio en cada punto, y la normal en la superficie apunta en la dirección del vector de gravedad.

Si bien el geode representa mejor la forma de la tierra, para mayor simplicidad, en este trabajo, se usa el elipsoide como la superficie de referencia.

En la siguiente figura se representan gráficamente las diferencias entre el elipsoide y el geode, se muestra un dibujo que muestra las diferencias entre el elipsoide, el geode y la altura del relieve.

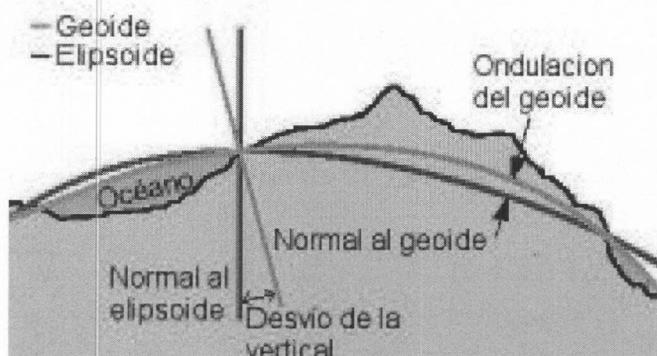


Figura 5.3: Diferencias entre el geode, el elipsoide y la altura del relieve.

El DEM

Lo que le falta al geode, para representar completamente la forma de la tierra, es el relieve de los continentes. Eso sólo es posible con un modelo de elevaciones (Digital Elevation Model - DEM). Más adelante se describe detalladamente el utilizado en este trabajo, aunque no es el único existente. Existen otros que varían en la resolución o el origen y precisión de los datos

5.2.2. Modelo WGS 84

Todo lo descrito en los puntos anteriores define un **marco de referencia**. Al darle valores a las variables de dicho marco, se empieza a hablar de **datum**. En este trabajo -y en todo el proyecto SAOCOM en general-, el

datum utilizado es el descrito en la definición del WGS84 (*World Geodetic System 1984*) [USG98]. En él, se describen los elementos que definen la tierra y que van desde la forma del elipsoide y el geoide, hasta el modelo gravitatorio.

- Elipsoide WGS84:

Parámetro	Valor
Semieje mayor (a)	6378.137 Km
Achatamiento (f)	1/298.257223563
Velocidad Angular (ω)	$7,292115 \times 10^{-5} \frac{rad}{seg}$

Cuadro 5.4: Definición del elipsoide WGS84

Además, dentro del marco de referencia, se define un sistema de coordenadas cartesianas, fijo a la tierra (ECEF), que es el descrito en 5.1.

5.2.3. Modelo de terreno

En el trabajo desarrollado, existe una funcionalidad requerida por el planificador que necesita conocer la altura del relieve de la superficie en cualquier punto. La única forma de saberlo, es teniéndolas guardadas de antemano en algún archivo que se pueda consultar al momento de necesitarlas.

Eso es el DEM: una matriz gigante que se accede por latitud y longitud, y guarda el valor de la altura del terreno para ese punto.

Existen varios modelos de estas estructuras que difieren en aspectos tales como el cubrimiento, la resolución, la proyección, etc. En este caso se utilizó uno denominado GTOPO30, cuyas especificaciones principales -y que importan al uso específico en esta aplicación- se describen en esta sección.

Los archivos se pueden obtener desde internet (ver [USG88]) donde puede ser descargado de forma gratuita. En el sitio web de la referencia, se encuentra la documentación de referencia con las especificaciones completas.

GTOPO30

El GTOPO30 es el resultado de la unión de datos de altitud del terreno de diferentes orígenes, para conformar una grilla a nivel global. Cumple con los requerimientos exigidos en la funcionalidad para la que se lo utiliza por lo que

fue integrado en la aplicación, mediante una interfaz para leer e interpretar los archivos. En los párrafos que siguen, se detallan los aspectos que interesan para el uso en el GEOSAR.

Especificaciones

La información viene en una grilla georreferenciada, de 30 segundos de arco de resolución (aprox. 0.00833333° , o 1km a la altura del Ecuador). El sistema de coordenadas vertical es la altura en metros sobre el Nivel Medio del Mar. El horizontal, es el correspondiente a las coordenadas geográficas referidas al geode especificado en el WGS84 (el datum). El cubrimiento es global (de -90° a $+90^\circ$ de latitud y de los -180° a $+180^\circ$ de longitud) lo que resulta en una matriz de 21.600 filas por 43.200 columnas. Las alturas varían en un rango que va desde los -407 a los 8752 metros. El mar está enmascarado con un valor de -9999, y las costas tienen un valor de 1 metro para distinguirlas del agua.

Estas características, en especial la resolución, hacen que ciertas areas de terreno como islas de superficie menor a 1 km^2 desaparezcan del mapa.

Presentación de los datos Semejante volumen de información es muy incómodo de manejar, por lo que todo el DEM esta fraccionado en mosaicos como se ve en la figura 5.4.

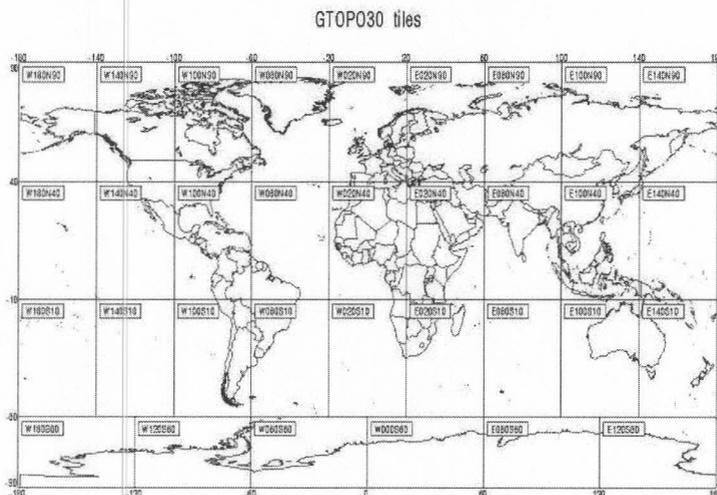


Figura 5.4: Mosaicos de división del GTOPO30

Para el rango de longitudes que va desde -60° a 90° , se divide en 27 sectores de 40° en longitud, por 50° en latitud. Entre -90° y -60° , se divide en 6 sectores de 30° en latitud, por 60° en longitud.

Cada una de estas piezas esta asociada con un conjunto de archivos del mismo nombre, -que hace referencia a las coordenadas de la esquina superior izquierda del mosaico-, y cuyas extensiones determinan la información que hay en él.

Por ejemplo, todos los archivos de nombre W180N90, contienen información referida a la superficie que va desde los -180° a los -140° de longitud, y desde los 40° hasta los 90° de latitud. El archivo W180N90.DEM, tiene las alturas correspondientes al mosaico descripto por el nombre.

La tabla 5.5 muestra los archivos asociados a un mosaico. Para un mayor detalle de alguno, ver la referencia [USG88].

Extensión	Descripción
DEM	Datos de altura
HDR	Información sobre la forma en que están guardados los datos
DMW	Información para georreferenciar el DEM.
STX	Valores máximo, mínimo, medio y el desvío estándar.
PRJ	Información de la proyección y el datum de los datos.
GIF	Imagen del área de cubrimiento.
SRC	Matriz que indica el código de la fuente usada para obtener el valor.
SCH	Información equivalente a la del archivo .HDR, pero para el archivo de estadísticas.

Cuadro 5.5: Extensiones de los archivos del DEM

Formato de los datos

Los datos de altura se encuentran en los archivos con extensión *.DEM*. Los que cubren 40 grados de longitud por 50 de latitud, contienen una matriz de 6000 filas por 4800 columnas, y los que cubren 30 grados de latitud por 60 de longitud, tienen 3600 filas por 7200 columnas. En ambos casos, la matriz está ordenada por fila, es decir, la fila n seguida de la fila $n + 1$. Los

datos de altura se representan con enteros de 16 bits guardados en formato binario, *big-endian* (el bit más significativo del número se almacena primero).

Precisión

La heterogeneidad de las fuentes que dieron origen al DEM, hace que la precisión varíe de un punto a otro en la superficie de la tierra. Así, aquellos que provengan de una fuente con poco error serán más seguros, mientras que los de fuentes sin un registro del error, serán más inciertos.

En general, los datos provenientes de formatos *raster* (DTED, USGS DEM) tienen mayor precisión que los provenientes de formatos *vector*. Por ejemplo, el DTED y el USGS DEM tienen un error de ± 30 metros con una confianza del 90%. Si el error se mide asumiendo una distribución Gaussiana con media 0, el RMSE (root mean square error) coincide con el desvío estándar de la distribución, y para este caso será de 18. En cambio, para un formato *vector* como el DCW, se calcula que el error lineal es de unos ± 160 metros, y el RMSE es 97.

La siguiente tabla muestra estos valores para las distintas fuentes de datos:

Fuente	Error Lineal	RMSE
DTED	30	18
DCW	160	97
USGS DEM	30	18
AMS maps	250	152
IMW maps	50	30
Perú map	500	304
N.Z. DEM	15	9
ADD	muy variable	

Cuadro 5.6: Tabla de errores según el origen de los datos

5.3. Proyecciones de mapas

La inclusión de distintas proyecciones, es producto de la necesidad de personalizar la vista que se tiene del mapa. No siempre la vista cilíndrica - default- es buena para operar con todas las regiones de la tierra. Por ejemplo, si se quiere obtener una imagen de alguno de los polos, lo mejor es tener

una proyección ortográfica (se muestra en la figura B.6) centrada en algún paralelo cercano al 90 (norte o sur).

Definiciones

En este contexto, una proyección, es un procedimiento matemático que mapea la superficie de un cuerpo en tres dimensiones -la tierra, en nuestro caso- a una superficie en dos dimensiones -el mapa.

Es decir, que una proyección es una función f , que toma como parámetros los valores de latitud y longitud, y devuelve, coordenadas (x, y) en un plano:

$$f : (\phi, \lambda) \rightarrow (x, y)$$

Dicha transformación siempre implica una distorsión, aunque se puede elegir una propiedad a conservar. Así surgen cuatro características de una proyección (que no pueden presentarse a la vez):

- **Conformalidad:** se dice que la proyección es *conforme* cuando preserva la forma de cualquier área, es decir, preserva los ángulos. En este tipo de proyección, los meridianos y paralelos, se cortan en ángulos rectos.
- **Equivalencia:** es equivalente cuando las áreas en una región del mapa son las mismas que las áreas en la superficie esférica. Para mantener esta propiedad se aplican distorsiones y la proyección ya no es conforme.
- **Equidistancia:** la proyección es equidistante cuando preserva la distancia entre ciertos puntos.
- **Dirección verdadera:** esta proyección conserva las direcciones entre los puntos y el centro del mapa. Se las denomina *Azimutales*.

Existen tres tipos de proyección:

- **Cilíndricas:**
la proyección se obtiene de “envolver.^a la tierra con un cilindro que es tangente a la superficie en un paralelo.
Ejemplos de esta este tipo son las proyecciones Mercator y Cilíndrica (figura 5.5).

Si el cilindro toca la superficie en un meridiano (figura 5.6), se obtiene otra proyección. Por ejemplo, la proyección Mercator Transversa o de *Gauss-Krüger*.

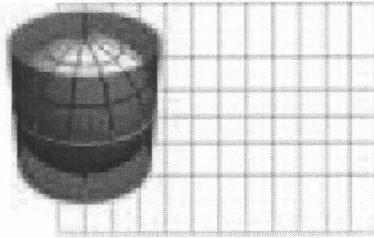


Figura 5.5: Proyección Cilíndrica

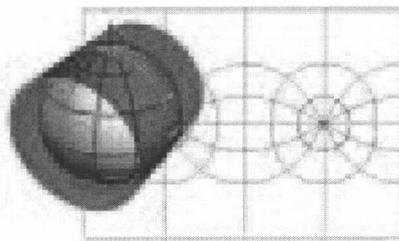


Figura 5.6: Proyección cilíndrica transversa

- Pseudo cilíndricas: se generan como las anteriores, salvo que las líneas de meridianos y paralelos están curvadas según los requerimientos de la proyección (cubrimiento, distorsión, continuidad, etc). Un ejemplo de este tipo es la proyección Goodes-Homolisine, que se muestra en la figura 5.7. En esta proyección, la deformación es mínima.

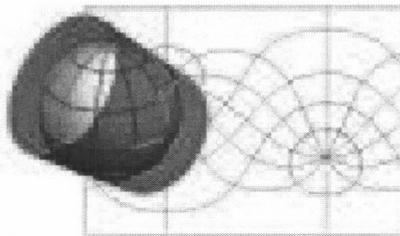


Figura 5.7: Proyección pseudo cilíndrica

- Cónicas: en este tipo de proyección, esfera está envuelta por un cono. Como resultado de esto se obtienen proyecciones como las de las figuras 5.9 y 5.8.

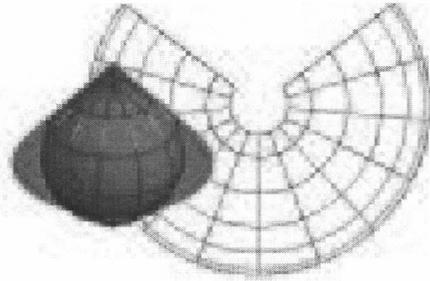


Figura 5.8: Proyección cónica

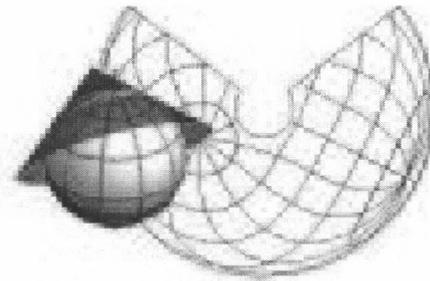


Figura 5.9: Proyección cónica oblicua

- Azimutales: En estas proyecciones, la superficie de contacto es un plano tangente a un punto. Las proyecciones azimutales conservan las direcciones verdaderas desde el centro del mapa, a cada punto de la proyección. Se presenta como ejemplo de estas, la proyección ortográfica (figura 5.10)

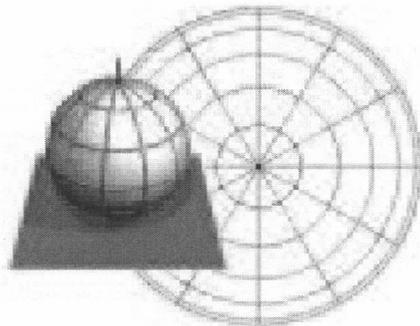


Figura 5.10: Proyección Ortográfica

El tipo de proyección a ser usada dependerá de la propiedad principal deseada y de la región a ser mapeada.

En esta sección se pretendió dar definiciones elementales sobre proyecciones de mapas. Aunque varias estén implementadas en la aplicación, se asume que el usuario tendrá cierto conocimiento sobre el tema, como para determinar las características de la proyección de uso según sus necesidades.

5.3.1. Proyecciones de mapas en GEOSAR

Como se dijo en la introducción a la sección, en el programa es posible cambiar algunos parámetros para personalizar la proyección del mapa. Esto se hace desde un menú que tiene todas las configuraciones del mapa: color, definición, paralelo y meridiano central, etc. Ejemplos de estas proyecciones, son las figuras que forman parte del apéndice B. Todas ellas fueron generadas desde el entorno GEOSAR, y es exactamente así, como se ven en la aplicación.

Capítulo 6

El satélite

Para este trabajo, solo resulta relevante modelar el satélite como un objeto orbitando, con la capacidad de transportar alguna carga útil. Para el desarrollo, solo fue necesario conocer las características de la órbita y algunas otras de la antena del radar, dejando de lado las cuestiones que hacen al funcionamiento del satélite en general (baterías, plataforma, etc).

En este trabajo, todos estos aspectos están aplicados al SAOCOM: a la órbita, sus modos de actitud y a la carga útil. De todos modos, las piezas del software desarrollado no son de funcionamiento exclusivo para ese satélite en particular y pueden utilizarse con otros. Por ejemplo, el propagador de órbita sirve para cualquier objeto del cual se conozca un TLE.

En esta sección, se describen las características para el caso del satélite SAOCOM.

6.1. La órbita del SAOCOM

Para esta misión, se ha diseñado una órbita heliosincrónica de 619km de altura promedio. Los datos que se muestran en la tabla que sigue fueron extractados del TLE usado para los tests del propagador. Estos valores pueden variar con el tiempo, pero el programa no deberá modificarse pues solo son parámetros del propagador.

Otras características importantes de la órbita, son:

Repetición:

Las características de la órbita hacen que cada 16 días repita el track sobre la tierra. Se combinan los valores de inclinación, la cantidad de

Elemento orbital	Valor
Inclinación	97.8892°
Ascensión recta del nodo ascendente	131.5371°
Excentricidad	0.0000100
Argumento del perigeo	90.0000°
Anomalía media	45.0000°
Movimiento medio	14.82154588

Cuadro 6.1: Parámetros orbitales del SAOCOM

revoluciones en el día, la cantidad de revoluciones del período de repetición y número de días de repetición, para que el punto subsatelital se repita cada 16 días exactamente.

6.1.1. Sistema de coordenadas de la órbita

Para definir posiciones u orientaciones respecto del objeto, es necesario definir un sistema de coordenadas centrado en el satélite. Este nuevo marco de referencia, dependerá de la posición y velocidad del vehículo, por lo que cambiará en cada instante de tiempo. Además, dependerá del método de alineación del satélite respecto de la órbita. En el GEOSAR, este sistema está fijo, y sigue la convención de *Alineación con el Nadir y la velocidad, en coordenadas inerciales*. Es decir que el eje Z de este nuevo sistema de coordenadas apunta en dirección al centro de la tierra (el vector de posición del satélite, en sentido inverso), el eje X es aproximadamente el vector velocidad, y el Y es ortogonal al plano determinado por el eje Z y el vector velocidad, donde los vectores de posición y velocidad están referidos al sistema inercial.

Entonces, los ejes se definen como:

- El eje \hat{x} : vector ortogonal al plano generado por z e y .
- El eje \hat{y} : vector ortogonal al plano formado por el versor z y la velocidad
- El eje \hat{z} : vector que apunta al centro de la tierra.

Los componentes del sistema son:

Siempre que se cuente con la posición y la velocidad, se podrá definir este sistema puesto que sus parámetros dependen de ellos. En nuestro caso, son los valores de retorno del propagador.

Origen	Centro de gravedad de la nave
Plano de referencia	El plano que contiene los ejes \hat{x} e \hat{y}
Dirección principal	Dirección del vector velocidad

Cuadro 6.2: Parámetros del sistema del satélite

Sea $P = (p_x, p_y, p_z)$ la posición del satélite, y $V = (v_x, v_y, v_z)$ la velocidad, para un instante t . Entonces, los ejes del nuevo sistema de coordenadas se obtienen mediante el siguiente procedimiento:

$$\begin{aligned}
 Z &= \frac{-P}{|P|} \\
 Y &= Z \times V \\
 X &= Z \times Y
 \end{aligned}
 \tag{6.1}$$

En la figura 6.1 estos ejes son los etiquetados como X_s , Y_z y Z_s . En dicha figura se muestra el sistema, posicionado respecto del sistema de coordenadas global (ECI).

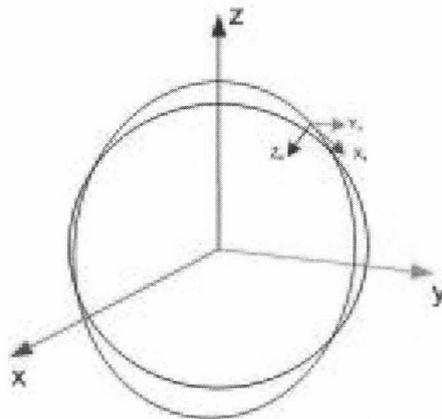


Figura 6.1: Sistema de coordenadas del satélite respecto de la posición en la órbita.

Este sistema de coordenadas se utiliza la sección 7.4.1 para calcular los ángulos de visibilidad de un punto en la superficie.

6.2. El Radar de Apertura Sintética (SAR)

Para el propósito para el cual fue pensado el GEOSAR, sólo interesan las características geométricas de la antena, y la forma en que se determina un apuntamiento y un haz.

Las características geométricas son la posición (se considera que la antena está situada en el origen de coordenadas del sistema definido en 6.1.1), y la orientación respecto del satélite. Más adelante se describe el sistema de coordenadas de la antena.

Respecto del modo de apuntamiento, los haces de la antena se representan mediante dos ángulos en rango -ver figura 6.2- (mínimo y máximo) que definen la zona del terreno que "mira" el radar. Se asume que la antena solo tiene capacidad de variar el haz en rango, por este motivo, el ángulo en azimuth estará fijo en 0° , o 90° , dependiendo de la convención adoptada. Como se puede ver en la figura 6.2, un haz válido tendrá la componente en azimuth en 90° . Estos ángulos, son en realidad abstracciones de un mecanismo complejo que trabaja con tiempos de emisión, espera y recepción, explicados en la introducción.

6.2.1. Sistema de coordenadas de la antena

La antena está inclinada en un cierto ángulo respecto del eje x del sistema de coordenadas del satélite. Este valor está fijo 33° pero puede cambiar. De todos modos, en la aplicación, este ángulo es un parámetro configurable.

La rotación de la antena, respecto del eje X del satélite -*tilt*- define un nuevo sistema de coordenadas. Si α es el tilt de la antena, entonces $RotX(\alpha)$ es la matriz que transforma las coordenadas del sistema del satélite de la antena. Por lo tanto, si M es la matriz calculada con el procedimiento descrito en 6.1.1, entonces:

$$A = RotX(\alpha) \times M$$

transforma coordenadas, del sistema inercial, al sistema de la antena; y si C_e es un punto (x, y, z) en coordenadas ECI, entonces, $C_a = A \times C_e$ es el mismo punto, pero en coordenadas de la antena.

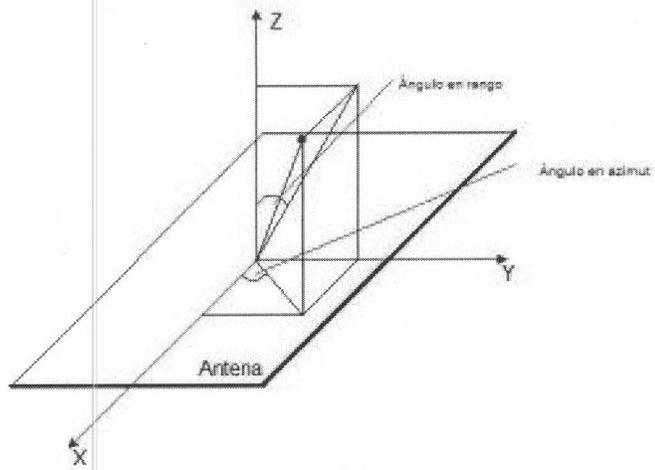


Figura 6.2: Sistema de coordenadas de la antena con ángulos en rango y azimut.

Capítulo 7

Descripción del software GEOSAR

7.1. Introducción

En esta sección se detallarán las características de la aplicación desarrollada, desde el punto de vista funcional y el entorno de trabajo; dejando de lado las cuestiones de implementación, que serán abordadas más adelante en el capítulo 8 . También se ampliará la información sobre el contexto de uso (funcionalidades, entorno y usuarios a los que está dirigida) y los posibles cambios y ampliaciones más probables.

7.2. Presentación

La aplicación se utiliza en un entorno de ventanas. Tiene una interfaz gráfica interactiva, con una ventana principal donde se muestra un mapa con el *escenario* de la misión. En él se visualizan los recursos (el satélite, las estaciones, las capas de información, etc.) y todos los objetos que se generen con las herramientas que provee el sistema.

Estos objetos -las adquisiciones, representadas por el *groundtrack* y el correspondiente *swath*; las regiones de interés, etc.- son editables mediante otras herramientas que también se incluyen (ventanas de edición, de simulación de la pasada, etc).

En definitiva, el sistema permite definir un escenario con recursos (satélite y estación terrena), sobre los cuales se pueden hacer operaciones para generar otros objetos (pasadas, adquisiciones, etc.) y además se puede enriquecer este

escenario, con regiones de interés definidas de forma interactiva, o importando capas de información en formatos estandar y propios.

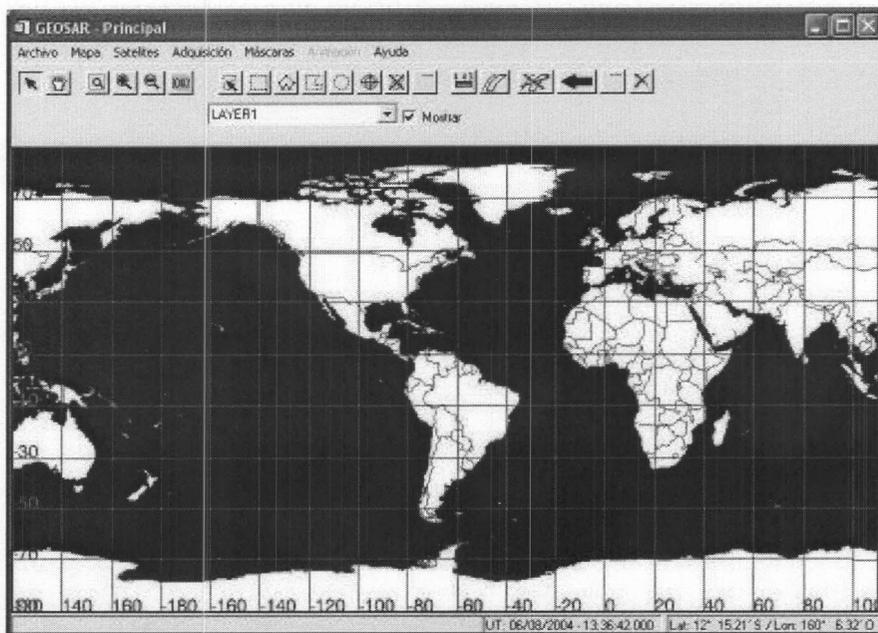


Figura 7.1: Ventana principal del GEOSAR

7.3. Contexto de uso

Como se explicó en la introducción, este paquete de software, es una pieza más dentro de uno más grande cuya funcionalidad es asistir a la planificación y a la ingeniería del instrumento SAR. En ese contexto, el GEOSAR era originariamente, un asistente al planificador de adquisiciones. Luego fue adquiriendo capacidades que la convirtieron en una herramienta con sentido en sí misma. Este crecimiento hizo que la operación de la aplicación pueda darse en ambientes diferentes, conviviendo con otras aplicaciones u operando sola, y con diversos tipos de usuarios.

7.3.1. Usuarios

En las tareas de planificación, la herramienta se utilizará como un asistente al planificador de adquisiciones. En este caso, el usuario -que es el op-

erador de la estación terrena- está familiarizado con los conceptos presentes a lo largo de la interacción con el software GEOSAR. Maneja las nociones asociadas con órbitas (en particular conoce la del SAOCOM), y conoce en detalle las características del instrumento.

Además será necesario que esté al tanto de las características propias del programa en sí, como ser: el modelo de propagador (se deberá actualizar la base de datos de TLE) con sus limitaciones y el error en sus *outputs*, las especificaciones del modelo de tierra, el modelo de elevaciones, las proyecciones de mapas, etc. Todos estos, son elementos conocidos para este usuario pues son de uso frecuente en este tipo de software.

Es importante notar que en modo asistente, el usuario tendrá la ilusión de que el planificador y el GEOSAR son una misma cosa puesto que este último se iniciará solamente con un click en un botón de la pantalla del planificador, sin necesidad de procedimientos adicionales.

En el caso de que sea otra persona la que opera el programa, también necesitará saber acerca de órbitas -en particular deberá conocer la del SAOCOM-, las prestaciones del satélite (modos de actitud, capacidades de transmisión, etc) y el instrumento (características de los haces, capacidad de operación, etc).

Este modo de operación es posible debido a las características del programa para su uso en modo *standalone*. El GEOSAR se puede utilizar en tareas como la creación *offline* de layers de información, escenarios de misión simulados, etc. De todos modos, las características propias del instrumento no se encuentran dentro del dominio de aplicación del GEOSAR, sino en otro paquete. No obstante, los modos, y las características de los haces son necesarias para poder utilizar el programa de forma realista (para no generar adquisiciones con una duración mayor que el máximo posible).

7.3.2. Software

Se ha dicho que el GEOSAR actúa de modo *standalone* o como un asistente al planificador. En el primer caso el ambiente de software es trivial, encontrándose la aplicación trabajando de forma individual. En el segundo caso, el planificador interactúa con el programa por medio de comandos que se envían mediante archivos. Si bien este modo de interacción se explicará en detalle en una sección más adelante, existe otro modo de interacción.

Se trata de el intercambio de fragmentos de código, que se insertarán en el

simulador del instrumento SAR ¹. Este modo de interacción se incluye en este informe, pues se intercambia una pieza esencial en el trabajo. Se trata de la clase que implementa el propagador, que se usa para obtener datos de posición y velocidad.

El otro modo de interacción con el simulador, se da al calcular los rangos del swath, pero se hace por intermedio del planificador. Esto se explica más en detalle en la sección 7.6.

En general, el GEOSAR, tiene la capacidad de funcionar como asistente independientemente del software que se encuentre del otro lado. Esto se debe a que si bien este trabajo consistió en el desarrollo de una aplicación particular para el SAOCOM -y más precisamente para el SAR-, fue concebido con la intención de servir como una herramienta capaz de recibir consultas y responder mediante una interfaz definida (que ahora es por archivos pero puede ampliarse y manejar *sockets*). Cualquier otra aplicación que necesite servicios de cálculo geométrico con el satélite (y la órbita) y objetos en la tierra, puede hacer una consulta mediante un mensaje especificado, y recibir la respuesta.

7.4. Funcionalidades de la aplicación

Como se dijo en el punto anterior, GEOSAR se pensó inicialmente como un asistente para el planificador, como un simulador de tierra y órbita. Luego, evolucionó en una aplicación con capacidades propias, conservando las originales. La idea de este desarrollo es obtener una herramienta que facilite un principio de planificación de las misiones. Que pueda determinar de forma temprana la posibilidad y la conveniencia de alguna adquisición, en función del área de interés y del momento en que se quiera realizar.

Para cumplir este objetivo, se desarrollaron un conjunto inicial de herramientas de aplicación específica para visualización de áreas de interés, y dibujo de las pasadas. Estas herramientas se corresponden con funcionalidades, que se describen a continuación en lo que sigue.

¹Este simulador, junto con el Planificador, forman parte del paquete de software en el que se incluye el GEOSAR

7.4.1. Cálculo de ángulos de pasada

La idea de esta funcionalidad es calcular el ángulo con el que la antena “mira” cierto punto de la tierra en una ventana de tiempo dada.

Para llevar a cabo el cálculo, son necesarios tres parámetros obligatorios, más uno opcional que depende de los requerimientos del usuario. Se necesita delimitar el área de interés, sobre la cual se quiere hacer la adquisición. Dicho área puede ser un punto en el terreno (un par de coordenadas de latitud y longitud) o el centro de una región definida como un polígono.

El centro de una región se define como:

Sea un polígono definido dentro del área $(lat_{min}, lon_{min}, lat_{max}, lon_{max})$, entonces el centro será el punto (ver figura 7.2):

$$lat_c = \frac{(lat_{min} + lat_{max})}{2}, \quad (7.1)$$

$$lon_c = \frac{(lon_{min} + lon_{max})}{2} \quad (7.2)$$

$$(7.3)$$

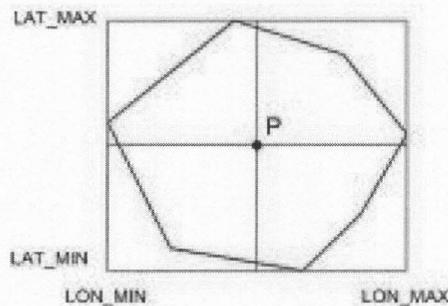


Figura 7.2: Punto (lat_c, lon_c) en una región poligonal.

También se debe informar una fecha a partir de la cual se podría hacer la adquisición. No necesariamente es la fecha en que se “prende el radar”, sino el momento en que se comienza a propagar la órbita para calcular los ángulos. Este es un requerimiento que depende de variables como la urgencia del cliente o las posibilidades del satélite.

Un ejemplo sería: “sacar una foto dentro de quince días”, o “en algún momento a partir de hoy”, o directamente indicando una fecha.

El tercer parámetro es la fecha límite para tomar la foto. Por ejemplo, “en menos de 5 días”. Éste es el parámetro opcional, puesto que por más que no haya una fecha límite, no tiene sentido propagar más allá del período repetición.

El último dato necesario es la resolución temporal. Indica el paso de tiempo en la propagación de la órbita. Cuanto mayor sea, menor será el tiempo de cálculo, pero la pasada que se obtiene es menos precisa. Si se considera que la velocidad del satélite es de $7 \frac{km}{s}$, en solamente 10 segundos, el satélite puede pasar por encima de la región de interés, sin detectarla. A diferencia de cuando se dibuja el swath, en esta operación, este parámetro sí es crítico.

En base a este conjunto de parámetros, se determinará otros como la fecha de inicio de la adquisición -UT inicial- y la duración. Los ángulos que se obtienen de este punto, se filtran en función de los haces posibles. Luego del cálculo, se muestra una tabla con los resultados obtenidos. Cada fila contiene tres columnas con una fecha, un ángulo y si la pasada es ascendente o descendente. La fecha es el instante en el que el satélite mira el punto con el ángulo de la siguiente columna. El nodo es la dirección de la pasada.

Esta operación puede llevarse a cabo desde la interfaz del GEOSAR, o mediante una consulta externa, por medio de un comando por intercambio de archivos.

Detalles del cálculo

El problema consiste en: dado un instante y un punto de la tierra, decidir si es visible para la antena del satélite.

En este contexto, *visible* significa que el punto se encuentre dentro de un rango de ángulos válidos. Es decir, con un ángulo en azimut cercano a 0 (dentro de una tolerancia configurable), y un ángulo en rango tal, que esté sobre la superficie terrestre, en la cara visible en ese momento.

Básicamente, se obtiene el vector al punto, en coordenadas del satélite, para cada instante de la pasada -hay que tener en cuenta que cada punto de la órbita define un sistema nuevo- y se calcula las componentes en rango y azimut del ángulo absoluto.

El cálculo consiste de 4 etapas:

1. Transformar el punto a coordenadas del satélite, utilizando la matriz

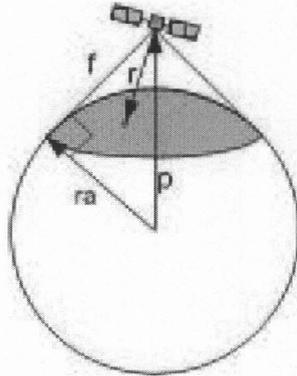


Figura 7.3: Cara visible de la tierra

de transformación que se obtiene mediante el precedimiento del punto 6.1.1.

2. Calcular las componentes en rango y azimut.
3. Filtrar los ángulos y eliminar los que no se encuentren dentro del cono de visibilidad del momento (ver figura 7.3).
4. Filtrar los ángulos con componente en azimut mayor que cierto umbral.

Estos pasos corresponden a un instante de tiempo. Si se quieren los ángulos para una ventana de tiempo, se deben aplicar para cada posición del satélite, en cada instante definido en la discretización. El output de la función, son los instantes tales que los ángulos y distancias al punto de referencia, superan los filtros propuestos.

En la figura 7.4, se ven los puntos que se buscan, y los puntos que no son válidos.

7.4.2. Dibujo del *swath*

Luego de determinar los ángulos útiles, hay que verificar el área de cubrimiento de las posibles adquisiciones. En este punto se imprimen sobre el mapa la huella y el *swath* para los haces elegidos. Esto se hace mediante una ventana en la cual hay que setear algunos parámetros como UT de inicio, duración, resolución temporal y ángulos mínimo y máximo. El UT de inicio y los ángulos, se pueden obtener a partir de los calculados en el punto anterior. La

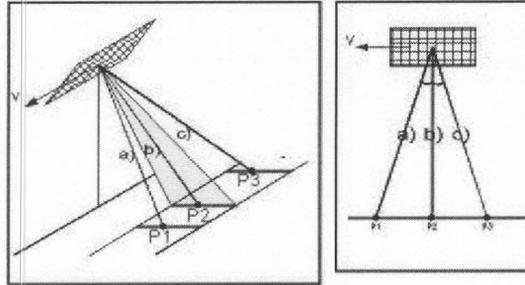


Figura 7.4: Tres puntos y su visibilidad desde la antena. El único válido es el b)

duración dependerá de qué es lo que se desea en ese momento.

En este caso también es necesario definir la resolución temporal de la simulación. Pero este no es un parámetro crítico, puesto que en el mapa, el dibujo del swath da la sensación de continuidad, aunque la simulación (y los pasos del satélite) sea más gruesa.

Para calcular el groundtrack del satélite basta con transformar los datos de posición de coordenadas inerciales a coordenadas geográficas. En cambio, para calcular el swath, es necesario saber el punto de intersección entre un vector que va desde el satélite, con cierto ángulo en rango, y el elipsoide de referencia. Este procedimiento se detalla más adelante, en 7.4.3, donde se describe la función *Intersecar_elipsoide*.

Estos gráficos de pasada son editables en el entorno de GEOSAR. Así se pueden hacer ajustes de los parámetros UT de inicio y duración de los swaths para ajustarlos al área de interés. También se puede hacer un seguimiento paso a paso por cada punto de la discretización.

7.4.3. Cálculo de rangos

El problema consiste en calcular la distancia desde uno o varios puntos en la órbita, a tres puntos en la superficie de la tierra: nadir, borde cercano y borde lejano (ver figura 7.5), que se definen a partir de 2 ángulos medidos desde el nadir (la dirección que apunta en el sentido del eje Z del sistema de coordenadas del satélite).

En la práctica, estos ángulos serán los que definen un haz de la antena.

Este problema se resuelve para dos casos: calculando a) la distancia al

elipsoide, o b) al punto sobre la superficie terrestre.

- a) Para este problema, sólo se debe aplicar una función que calcule la intersección de una elipsoide con una recta. Ése es un cálculo cerrado que se explica más adelante. Se implementó en la función llamada *Intersecar_elipsoide*, que recibe un punto -o un vector de puntos- y un ángulo, y devuelve el vector de la proyección en el elipsoide, con su longitud. En este caso particular se debe llamar con los ángulos 0 -para el nadir- mínimo -para el borde cercano- y máximo -para el borde lejano. Al finalizar la rutina, en el parámetro de la longitud del vector de proyección, se encuentra la distancia buscada.

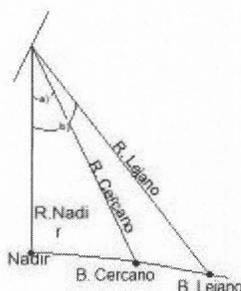


Figura 7.5: Los tres puntos (nadir, b. cercano y b. lejano) sobre el elipsoide.

- b) Este otro problema es un poco más complejo pues no tiene solución analítica. Se resuelve estirando el vector unitario que apunta en la dirección pasada como parámetro y testeando con el DEM si dicho punto se encuentra sobre la superficie de la tierra.

Algoritmos y métodos de cálculo

Para obtener los resultados de los puntos anteriores se implementaron los algoritmos que se detallan en los párrafos que siguen.

Intersección de una recta y el elipsoide

En particular, para resolver el problema principal es necesario poder determinar, dada una posición en la órbita y un ángulo, el punto en el elipsoide que resulte de estirar un vector del sistema de coordenadas del satélite, con el ángulo pasado como parámetro. Dicho problema se puede resolver de forma

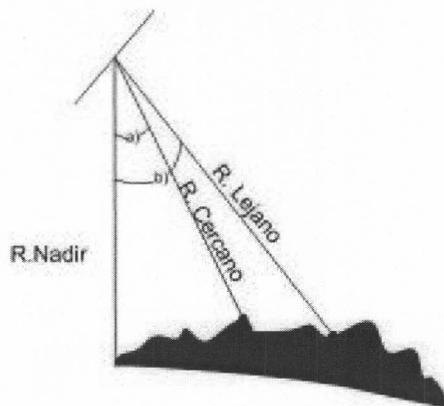


Figura 7.6: Los tres puntos (nadir, b. cercano y b. lejano) sobre la superficie.

analítica como se explica a continuación:

Sea α un ángulo, entonces v es el vector $(0, 0, 1)$ de las coordenadas del satélite, transformadas en coordenadas inerciales, y rotadas α radianes respecto del eje \hat{x} . Esto es:

$$v = M.(0, \text{sen}(\alpha), \text{cos}(\alpha))$$

donde M es la matriz de transformación del sistema de coordenadas del satélite al sistema de coordenadas inercial.

Geoméricamente, el punto buscado sería $p = x_s + d.v$ (donde d es un factor que estira el versor v , y x_s es el vector de posición del satélite en coordenadas inerciales). Entonces, cada componente de p se calcularía como:

$$x_p = x_s + d.v_0 \tag{7.4}$$

$$y_p = y_s + d.v_1 \tag{7.5}$$

$$z_p = z_s + d.v_2 \tag{7.6}$$

Como x_p está sobre el elipsoide, cuya ecuación es conocida, se debe cumplir que:

$$\frac{x_p^2 + y_p^2}{a^2} + \frac{z_p^2}{b^2} = 1^2 \tag{7.7}$$

donde a y b son los ejes mayor y menor del elipsoide.

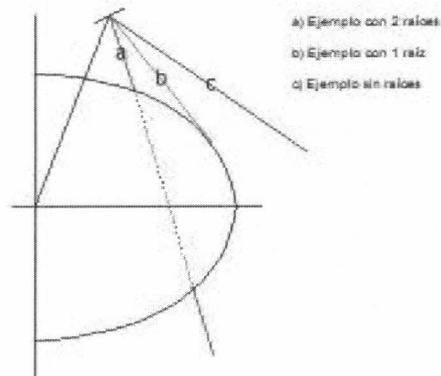


Figura 7.7: Los tres casos posibles en el problema de intersección de la recta y el elipsoide.

Luego, en la ecuación 7.7 se pueden reemplazar x_p, y_p, z_p por las ecuaciones 7.4, 7.5 y 7.6 para lograr que d (el factor del versor) sea la única incógnita, y reducir el problema al de hallar las raíces de un polinomio de grado dos, lo cual es fácil de resolver.

Al encontrar las raíces pueden pasar tres cosas, como se muestra en la figura 7.7:

1. obtener dos raíces reales positivas. Estas representan los dos cortes en el elipsoide, en este caso hay que optar por la de menor valor absoluto (el otro coeficiente estira el vector v hasta el lado no visible de la tierra).
2. obtener una sola raíz positiva. Lo que significa esto es que existe un solo corte. El ángulo es tal que al estirar el versor, éste es tangente a la elipse.
3. no obtener raíces reales. En este caso, no hay intersección con el elipsoide. Esto se puede interpretar como un ángulo inválido

El algoritmo del programa resuelve esta ecuación cuadrática y actúa según los tres casos enumerados.

En este caso se mostró la resolución para un solo punto. Como IDL es un lenguaje vectorial, este cálculo se hace para todos los puntos de la pasada al mismo tiempo, ayudando a la performance del sistema en general.

Cálculo de rangos sobre la superficie del DEM

Para comprender mejor la forma de resolución de éste problema, se escribe el siguiente pseudocódigo, donde α es el ángulo, pos es el punto de la órbita en coordenadas inerciales (ECI), t es el tiempo de referencia de pos y $rango$ es la variable donde el procedimiento retorna el valor.

1. $fin = FALSE$
2. $r_f \leftarrow Intersecar_elipsoide(\alpha, pos) + k$
3. $v \leftarrow RotarX([0, 0, 1], \alpha)$
4. $r = r_f, r_i = 0$
5. **repetir**
 - a) $v_n = r * v$
 - b) $(\psi, \phi, h) \leftarrow Convertir_LLA(v_n)$
 - c) **si** $|DEM[\psi, \phi] - h| < 0,01$ **entonces** $fin = TRUE$
 - d) **sino**
 - 1) $\frac{r=(r_i+r_f)}{2}$
 - 2) **si** $h > DEM[\psi, \phi]$ **entonces** $R_i = r$ **sino** $R_f = r$
6. **hasta fin**

Donde:

- *Intersecar_elipsoide*: es la función descrita en 7.4.3
- *RotarX*(v, α): rota al vector v en un ángulo α respecto del eje X .
- *Convertir_LLA*(v_n): convierte un vector en coordenadas del Satélite, a coordenadas Geográficas.
- *DEM*[i, j]: es la altura del terreno para el punto de latitud i y longitud j .
- k : es un parámetro regulable para manejar situaciones en que la altura del terreno se encuentre por debajo del elipsoide. Según las especificaciones del DEM (ver 5.2.3) el valor mínimo de una altura es -407 metros, entonces habría que estirar el vector lo suficiente como para considerar todas las posibilidades. Una posibilidad es sumar 500 m a la componente z del vector v (sumando alguna cantidad necesaria a las otras componentes para mantener la dirección del vector).

7.4.4. Capas de información

El mapa que se muestra en pantalla puede ser enriquecido con información que se genere desde el entorno, o desde un software externo. Para lograr esto se implementaron funcionalidades propias de dos tipos de software distintos:

- Graficadores

GEOSAR cuenta con un set de herramientas básico para generar gráficos al modo de los graficadores. Es decir, utilizando el puntero del mouse para dibujar polígonos, líneas o puntos. Estas formas se interpretan como regiones georeferenciadas en el mapa que cambiarán de aspecto convenientemente con la proyección elegida (ver 5.3). Además, cuentan con atributos gráficos (color, sombreado, etc) y con atributos propios relacionados con el objeto que representan (posición, descripción, etc), que pueden ser personalizados.

- Herramientas GIS

También se implementaron, algunas características propias de los GIS. Al ver el mapa es natural pensar que éste puede ser un sistema GIS. Y, de hecho, algunas características clave, están implementadas.

Hay un mapa georeferenciado con una estructura de capas que se pueden ocultar o mostrar, más cierto cálculo con la información adjuntada a él.

Los objetos dibujados en el mapa, se almacenan en forma de secuencia de coordenadas geográficas unidas por un vector de conectividad. Este método se corresponde con uno de los formatos usados en el software GIS: el formato *vector*. Ciertos formatos de archivo como los implementados en esta aplicación (*shp, dxf*) también utilizan este método. De hecho, la interfaz con estos sistemas es a través de estos dos formatos de uso frecuente. Con ellos se pueden importar capas generadas con software de GIS, o incluso exportar las hechas con GEOSAR -con alguna pérdida de información en el cambio de formato- para el análisis con otra aplicación.

La otra forma de almacenar información se conoce con el nombre de *raster*. A diferencia del *vector*, este formato guarda los datos en una grilla que se amolda a la proyección en uso. Un ejemplo de este formato son los archivos del DEM (usados para calcular los rangos) o cualquier imagen.

7.4.5. Persistencia del entorno

Como ya se comentó en la sección anterior, existe en la aplicación un mecanismo para salvar el ambiente de trabajo. Con esta funcionalidad se podrían mantener ciertas adquisiciones de rutina guardadas de antemano, lo cual sería un ahorro considerable de tiempo de planificación, mientras que los cambios en la órbita no invaliden el trabajo.

El escenario guardado, puede ser recuperado de dos formas posibles: La primera, cargándolo completamente desde cero. Es decir, que se elimina lo que haya en el mapa y se agrega lo nuevo; los layers hasta ese momento se eliminan y se generan los guardados. Las configuraciones del mapa y la grilla también se recuperan.

La otra forma, es insertando en el entorno actual, todas las capas del que se está recuperando. Lo que se hace es apilar todos los layers del entorno guardado sobre el último del entorno actual. El resultado es la suma de los dos escenarios. En este caso, la configuración del mapa y la grilla del entorno guardado, no se tienen en cuenta.

Es importante notar que el mapa y la grilla son independientes de los objetos del escenario. Es decir, que si uno cambia la configuración del mapa (por ejemplo la proyección), los entornos guardados siguen siendo compatibles independientemente de la proyección con que fueron guardados.

Otra forma de persistencia, aunque sea gráfica, es posible, exportando imágenes en formatos gráficos(*BMP, JPEG, etc*) a archivos, o al *clipboard*. También exportando a formatos GIS como *shp*.

7.5. Ejemplo de ejecución

A continuación se presenta un ejemplo de uso de las funcionalidades descritas en la sección anterior. En el caso planteado, hay que decidir si es posible llevar a cabo el requerimiento de un cliente. Entonces, la tarea consistirá en propagar la órbita, verificar qué áreas son visibles, y decidir en función de los resultados obtenidos.

Concretamente, el pedido que se hace es: sacar una imagen del área de la base Marambio, entre el 21 y el 22 de setiembre de 2004.

La secuencia de pasos se describe en la figura 7.8. En ella se ven las tres columnas con los actores que ejecutan la acción y los eventos que generan en

los otros.

En el apéndice B se muestra una serie de imágenes donde se puede ver la interfaz gráfica del programa en cada uno de los pasos.

Para mejor comprensión del lector o usuario, se describe cada paso y se hace referencia a la imagen en el apéndice.

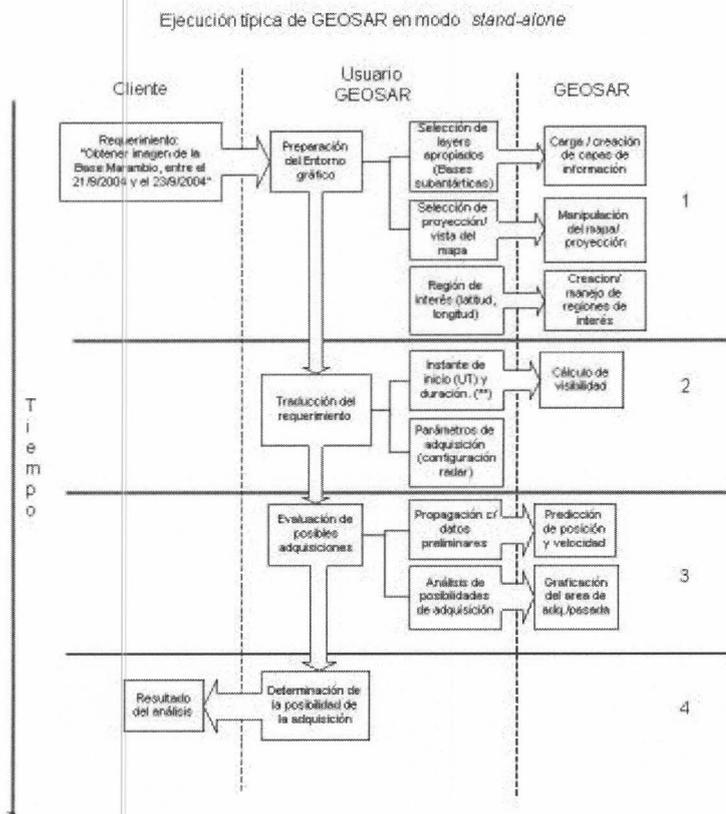


Figura 7.8: Ejecución modo Stand-alone

En el esquema se ve cómo se llega de los requerimientos de alto nivel del usuario, hasta completar el análisis de la posibilidad de adquisición. Para mejorar la descripción del ejemplo, el caso fue dividido en 4 etapas determinadas por las acciones del usuario de GEOSAR.

1. En la primer etapa, el usuario recibe el requerimiento en lenguaje coloquial, de parte del cliente. Es él quien debe determinar el área a adquirir

y es esa una de las ventajas de GEOSAR ya que ingresar un área de interés por medio de un conjunto de coordenadas es una tarea interactiva y simple. Es en este punto además, donde se lleva a cabo la operación con el mapa, personalizando la proyección, el centro, la vista y la aproximación (zoom). Además, en esta etapa se puede apelar a los layers y características de software GIS.

En el ejemplo, por tratarse de uno de los polos, se eligió la proyección *Stereographic* centrada en el meridiano 62° O (Argentina) y el paralelo 90° S (el polo sur). Este proceso se detalla en las figuras B.2 y B.3. Las figuras B.4 y B.5 muestran otros reacondicionamientos de la vista del mapa.

Ahora es turno de utilizar una capa de información. En este caso se inserta una generada con el mismo GEOSAR, que incluye las bases subantárticas. Esta capa se puede guardar en un archivo y utilizarse en el futuro. Si el requerimiento fuese obtener una imagen de, por ejemplo, Bariloche, entonces se podría cargar un mapa con ciudades argentinas o división política para localizar esta ciudad más fácilmente.

En la figura B.6 se observan las bases en el mapa. Por ser regiones generadas con GEOSAR (no provenientes de archivos Shape), se pueden editar y cambiar las propiedades de estas bases (figura B.7), por ejemplo, los atributos gráficos.

2. Una vez configurado el mapa de la forma más conveniente, y ya en la etapa 2, es momento de usar la funcionalidad que determina la visibilidad de la región: el cálculo de ángulos de visibilidad (figura B.8).

Pero para esto, hay que determinar algunos parámetros que dependen de los requerimientos. Estos son, el instante de inicio de propagación, que en este caso es el 21/9/2004, y la duración de la propagación. En este ejemplo, el requerimiento indica que la foto debe ser anterior al 23/9/2004, por lo que este parámetro será 2 días.

Con estos datos GEOSAR calcula con que ángulos en rango pasa el satélite, considerando que el de azimut debe ser muy chico. El resultado, es una lista con la fecha, el ángulo y el nodo (ascendente o descendente) de la pasada (figura B.8). En función de estos parámetros, se decide elegir 2 opciones que parecen convincentes (una de ellas por ejemplo, el 21/9/2004 - 10:34:22). En este punto, el usuario debe tener un conocimiento mínimo de los rangos de operación del SAR. Sino podría elegir una pasada donde la región esté “demasiado lejos”.

También habría que considerar la actitud del satélite, para elegir un haz que “mire.” a derecha o a izquierda.

Con esto se tienen los parámetros de adquisición necesarios para dibujar una pasada con swath incluido.

Como se verá más adelante, en el modo asistente se puede configurar el SAR de forma detallada. En este momento, no es eso lo que se quiere hacer.

3. El paso siguiente es graficar el swath en función del UT de inicio, la duración (ahora sí, de la pasada) y los ángulos del radar. Esta tarea implica propagar la órbita para los períodos seleccionados en el punto anterior, y dibujarla. Ambos pasos se hacen en uno solo, desde la ventana de propagación de la órbita. En la figura B.9 se ven las ventanas de cálculo de ángulos y de propagación. Esto es porque desde la primera se puede llamar a la segunda, eligiendo una fila (un tiempo y un ángulo) y con el botón “Propagar”. Esta acción setea algunos valores de la ventana de propagación, que el usuario puede modificar a conveniencia.

Se eligen graficar 2 pasadas. Al verlas en el mapa, es evidente que el tiempo de inicio de propagación correspondía exactamente con el del “sobrevuelo” de la región. Por eso, en este punto se puede hacer algún ajuste de fecha de inicio o duración con la ventana de edición de las pasadas (figura B.10). En esta ventana se encuentran todos los valores relacionados con la pasada, de los cuales se pueden modificar algunos (como los ángulos del haz) y ver instantáneamente el cambio reflejado en el dibujo.

4. El último paso es una operación del usuario donde decide si la adquisición es posible en base a los resultados de las etapas anteriores. En este caso se elige la pasada que se ve en la figura B.11.

En este modo de operación es donde más funcionalidades se utilizan. El entorno desarrollado provee de facilidades para la configuración del entorno gráfico, como ser posibilidad de zoom o de movimiento de la vista.

En la preparación del entorno de trabajo se puede hacer uso de las capas de información, las vistas del mapa, etc. Más adelante se muestran mapas de ejemplo, que son el resultado de sumar capas

7.6. Modo asistente al planificador

La aplicación tiene dos modos de uso que coexisten. Uno -el modo *stand-alone*- ya fue explicado. A continuación, se describe el otro modo: el de asistente al planificador.

En ambos, el programa es completamente funcional, y la diferencia es el foco de la atención. Mientras que en el primero de los modos, la idea es la evaluación de las pasadas y la interacción con el mapa, en el segundo, se configura el radar y se ajusta la adquisición, para lo cual el usuario debe conocer el radar en detalle.

En esta sección sólo se describirá el aspecto funcional de la interacción. Los detalles de la comunicación y cómo está implementada, se dejan para la sección 8.

7.7. Esquema de interacción

En la figura 7.9 se muestra cada actor en una columna, y la forma en que se comunican. Este esquema ayuda a la comprensión de las interacciones, y la secuencia en que se producen. Las flechas con texto (por ejemplo “orbita_ull”) indican el nombre del archivo donde se pasan los datos.

En el esquema se pueden ver las 3 etapas que forman el conjunto de las interacciones entre el GEOSAR y el planificador.

- 1) en este punto se calculan los ángulos de visibilidad para un punto seleccionado por el usuario. Esta etapa tiene una variante: se puede ingresar dicho punto directamente en la interfaz del planificador, o puede definirse en el entorno de GEOSAR, y seguir con el resto del procedimiento. Eso implica encender la interfaz de GEOSAR. Si bien esto puede parecer un poco costoso, la mayoría de las veces, el operador del planificador no sabe las coordenadas precisas del área que quiere adquirir. Además, es probable que luego de calcular ángulos se quiera graficar el haz definido. En este punto sí es necesaria la interfaz. Por esto, no siempre es inconveniente definir el área de esta otra forma. Esta variante se detalla en la figura 7.10.
- 2) en la segunda etapa, y al igual que en el modo *stand-alone*, la idea es ajustar el área alumbrada por la antena a la región de interés. Siempre

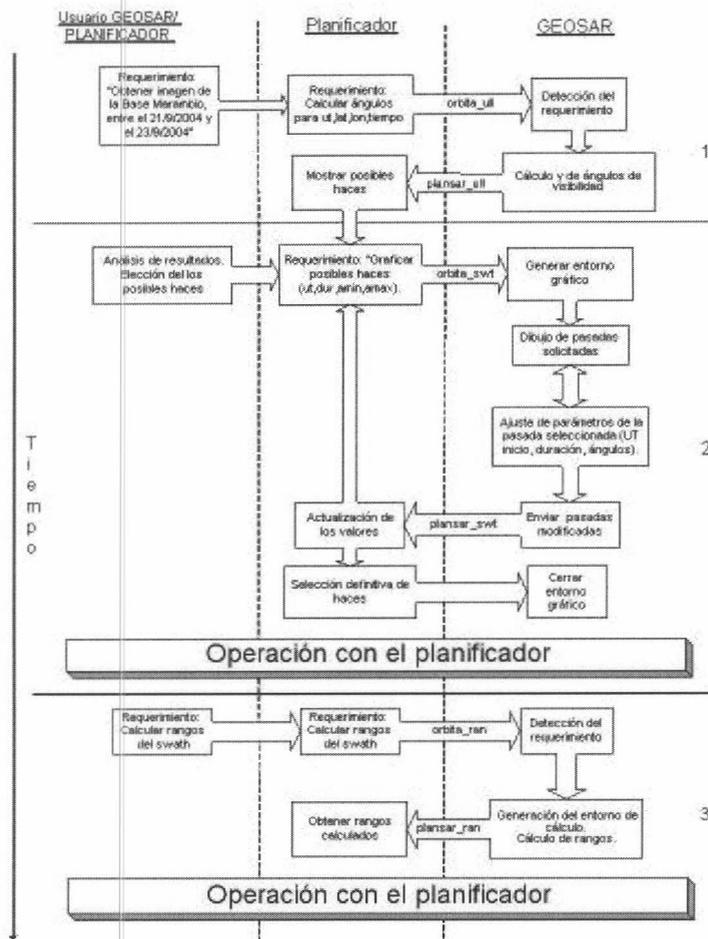


Figura 7.9: Esquema de interacción planificador-GEOSAR

está presente el objetivo principal de esta operación: detectar de forma temprana la conveniencia y posibilidad de una adquisición.

Esta operación de “ajuste” del swath, puede generar cambios en los parámetros que se pasaron inicialmente. Por eso, estos objetos modificados, pueden retornarse al planificador y mantener ambas interfaces actualizadas. Esta operación continúa hasta que el usuario decida que está conforme con los resultados obtenidos. Al finalizar, puede apagar la interfaz gráfica de GEOSAR pues no será utilizada otra vez.

3) La última de las interacciones corresponde al cálculo de rangos del

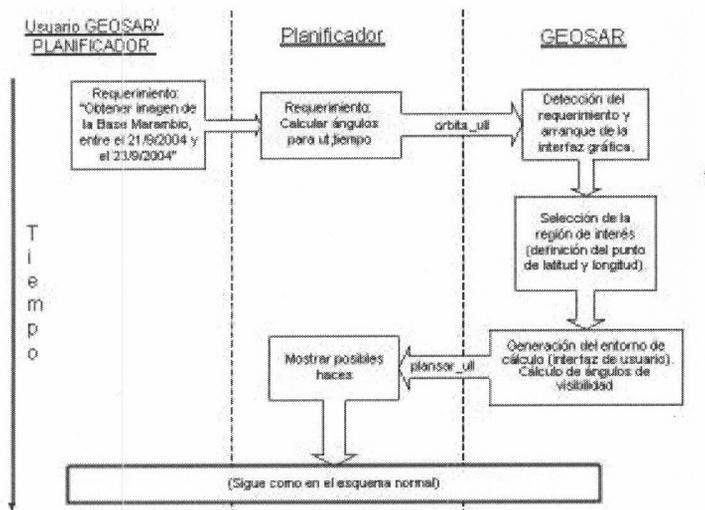


Figura 7.10: Variante en el esquema de interacción planificador-GEOSAR

swath. Este punto se explica con un poco más de detalle pues no fue comentado en la sección 7.4.

El radar necesita saber la distancia que lo separa de la región que está adquiriendo para calibrar parámetros relacionados con los tiempos de emisión y de espera de los pulsos electromagnéticos. Para ello, y porque dicha distancia también es producto de la relación entre la tierra y el satélite (y su órbita), es que se encomienda esta tarea al simulador de tierra y órbita.

Este modo de interacción es muy similar al primero puesto que sólo se hacen cálculos referidos a la posición del satélite con respecto a la superficie terrestre y no es necesaria la interfaz gráfica.

Como se mencionó antes, el objetivo es cubrir un requerimiento para la configuración de algunos parámetros para el radar: en este caso es necesario saber la distancia del satélite a tres puntos: nadir, borde cercano y borde lejano del swath en la superficie de la tierra.

Este cálculo puede hacerse de dos modos:

- Con respecto al elipsoide: en este caso se mide la distancia desde

la antena, hasta el elipsoide de referencia (WGS84 - ver 5.2.2).

- Con respecto a la altura del terreno: este otro caso es más complejo que el anterior. Aquí se mide la distancia con respecto a la altura verdadera de la superficie terrestre. Para ésto se utiliza un modelo digital de elevación (ver 5.2.3)

Actualmente están implementadas ambas funcionalidades y el usuario puede elegir con cuál de ellas hacer los cálculos. El motivo de tener dos implementaciones de la misma función es, simplemente, porque se utilizan para cálculos con necesidades distintas. En algunas circunstancias, no es necesario calcular la altura referida al DEM, y por eso es que se da la alternativa de utilizar el elipsoide. Incluso, existe implementada una opción para cambiar el elipsoide de referencia. Pero como el cálculo es, la mayoría de las veces, sobre el WGS84, no es conveniente utilizar esta funcionalidad si no se está seguro de lo que se quiere.

Los detalles de la implementación de estos algoritmos, puede verse en el punto 7.4.3

Capítulo 8

Detalles de implementación

8.1. El lenguaje utilizado: IDL 5.6

IDL es la sigla para *Interactive Data Language*. Se trata de un lenguaje multiplataforma, optimizado para el cálculo matricial, y orientado al desarrollo de aplicaciones de cálculo y visualización de datos, que brinda una gran biblioteca de funciones, y un marco de desarrollo que permite obtener aplicaciones de gran calidad -en cuanto a performance y a terminación en poco tiempo.

En este capítulo se da una breve introducción al IDL. Para un nivel mayor de detalle, consultar [Sys99].

8.1.1. Descripción

El entorno de programación se presenta como una ventana principal, que hace de editor de texto del código de IDL (además incluye la herramienta para generación de interfaces gráficas (GUIBuilder)). Se puede escribir un fragmento de código aislado -una función o un procedimiento-, compilarlo para incorporarlo al *top-level*, e invocarlo mediante un llamado desde la línea de comandos (incluída en la interfaz).

También se pueden generar proyectos. Se trata de conjuntos de archivos con definiciones de funciones, con un *entry point* desde el desde el cual comienza la ejecución. Para ejecutarlo se deben compilar todos los archivos `.PRO`.

Los *outputs* del IDL son:

- El código fuente -archivos de extensión PRO-: que se compila y ejecuta en una ventana de IDL

- Archivos de extensión SAV: que contienen el código precompilado en un formato de IDL. Las funciones definidas dentro de ellos, se pueden incluir en el entorno mediante una sentencia `RESTORE` ejecutada desde la línea de comandos. Éste es el modo común de exportar los proyectos desarrollados con esta aplicación.

En resumen, para ejecutar un programa escrito en IDL, es necesario incorporarlo al *top-level* (mediante un `RESTORE` o compilando el código fuente) y luego invocarlo desde la línea de comandos (todas operaciones realizables con lo menús y barras de herramientas de la interfaz).

Existe una tercer manera, que es utilizando el `IDL-runtime`. Esta es una aplicación que solo incluye el intérprete, y recibe como parámetro un archivo `.SAV`. En general, dicho archivo se corresponderá con un proyecto con un *entry point* desde el cual comienza su ejecución. En caso de no definirse uno explícitamente, el IDL busca una función del mismo nombre del archivo y la ejecuta. Si no la encuentra, dará un error.

A partir de la versión 6.0, se incluye una máquina virtual, que interpreta los archivos `.SAV`, que se puede descargar gratuitamente del sitio de RSI.

8.1.2. El lenguaje de programación

El lenguaje ofrece características para la programación imperativo y para la programación orientada a objetos. Es decir, que por un lado permite definir funciones y procedimientos, con todas las estructuras de control tradicionales (`if`, `for`, `while`, etc.); y por el otro permite definir clases y objetos, y relaciones de herencia (simple y múltiple) entre ellas.

La forma de definir los tipos de los datos también es muy simple y flexible (se definen cuando se inicializan), aunque esto a veces puede jugar en contra del programador, que no cuenta con ese tipo de verificaciones (como tampoco cuenta con, por ejemplo, los errores por la incorrecta definición de símbolos) por parte del compilador.

Programación orientada a objetos

Los objetos en IDL, son variables en el `HEAP`. Lo que implica que el programador tiene control total sobre el ciclo de vida (creación y destrucción). Se acceden a través de una referencia que son de acceso global.

El soporte de objetos que brinda IDL, provee de los siguientes mecanismos:

Clases e Instancias: Los objetos son instancias de una clase, que se define como una estructura de IDL: el nombre de la estructura es el nombre de la clase. Los datos de la clase, son los valores de la estructura. Una instancia de una clase, es una estructura de IDL, que se almacena en el HEAP y puede ser accedida mediante procedimientos y funciones especiales denominadas *métodos*, que se asocian a la clase.

Encapsulamiento: El encapsulamiento es la a capacidad de combinar datos y rutinas que afectan los datos en un único objeto. IDL logra esto al permitir el acceso a los datos, únicamente mediante los métodos de la clase. La información dentro de la estructura de la clase, no se puede acceder directamente (sólo dentro del contexto de la definición de la clase).

Métodos: Los métodos de las clases se pueden definir utilizando todas las herramientas de IDL. Las rutinas se asocian a una clase mediante una convención sintáctica en la forma de definir el nombre de la función:

```
(PRO | FUNCTION) < nombre_clase > : : < nombre_metodo >  
    , [param1] . . . , [paramn]  
    , [KW1 = kw1] . . . , [KWn = kwn]
```

Polimorfismo: Polimorfismo, es la capacidad de crear múltiples objetos de distinto tipo, que soporten un mismo conjunto de operaciones, Por ejemplo, muchos objetos gráficos soportan el procedimiento DRAW (que manda a graficar el objeto alguna salida especificada). La operación DRAW varía según el contexto. Es diferente enviar un gráfico a una impresora o a una ventana. El polimorfismo permite esconder los detalles de la implementación de cada tipo de operación.

Herencia: La herencia es la capacidad de un objetos, de heredar el comportamientos de otra clase de objeto. Es decir, que cuando se programen objetos operaciones similares, sólo se deberán programar las diferencias entre ambas. IDL soporta herencia múltiple. Es decir que un objeto puede heredar características de varios otros al mismo tiempo.

Persistencia: La persistencia es la capacidad de los objetos, de mantenerse en memoria luego de su creación. Los objetos persisten hasta su destrucción explícita o hasta que se destruye la instancia de IDL. La persistencia permite modificar los objetos, sin la necesidad de crearlos nuevamente. Por ejemplo, un objeto gráfico que contiene un PLOT puede

modificarse "on the fly" y visualizar los cambios, sin la necesidad de crearlo otra vez. En graficación directa, sería necesario ejecutar el comando PLOT con los nuevos parámetros.

Los objetos de IDL, pueden persistir en el tiempo aún después de concluida la sesión de IDL, guardandolos mediante los comandos SAVE y RESTORE.

GUIBuilder

El lenguaje incorpora una herramienta para editar ventanas de forma visual. Facilita la creación de las interfaces al permitir saber, en todo momento, el aspecto final de la ventana. También ahorra al programador, de escribir el código para generarla, y los mecanismos de obtención y clasificación de los eventos. Este asistente genera el código necesario para dibujar las ventanas, y permite definir prototipos de procedimientos para el manejo de los eventos de cada componente.

8.1.3. Manejo de memoria

IDL es un lenguaje orientado al cálculo matricial. Abstrae al usuario de los detalles en el manejo de memoria estática, encapsulando en el intérprete la gestión de la memoria. De esta manera permite generar vectores y matrices de gran tamaño de forma simple. Define varios modos de proyección de los los elementos (toda una columna $a[*],0$, un rango $a[2:30,*]$, etc), permite la concatenación de forma directa ($a=[1,2,3]$ y $b=[4,5,6]$, entonces $[a,b]$ será $[1,2,3,4,5,6]$). Esto permite mucha flexibilidad en el manejo de los datos, pero puede influir negativamente en la performance general de la aplicación.

Permite definir variables dinámicas (en el *heap*), y permitir generar estructuras dinámicas. Al no proveer de un mecanismo de *garbage collector* automático, el manejo de esta memoria es responsabilidad del usuario, y un mal manejo puede generar *leaks* de memoria, y la consecuente degradación de los recursos. Lo mismo sucede con los objetos, cuyo ciclo de vida (creación y destrucción) está a cargo del programador. Sí provee de un comando `HEAP_GC` que libera la memoria del *heap*, de las variables que no tengan referencias activas.

8.1.4. Motor gráfico

Graficación directa

La *graficación directa* es un método de graficación mediante comandos, que se basa en la idea de un dispositivo (*device*) sobre el cual se dibujan figuras por medio de comandos. Utilizando sentencias como `Plot`, `Map_set` o `Surface`, se pueden imprimir gráficos en una ventana. Este método, a pesar de ser más veloz que el de *Objetos gráficos*, es mucho más limitado y no tiene todas las ventajas de manipulación de los gráficos que sí tiene el otro. Concretamente, cuando se hace un `Plot`, los píxels en la pantalla no se pueden alterar. Para modificarlos, hay que modificar los datos y rehacer **todo** el dibujo (no solamente de ese plot). Con el otro método, se puede cambiar un solo objeto y redibujarlo. Esa es la razón principal para utilizar el método de los objetos. En general, se utiliza graficación directa sólo para una visualización elemental de los datos.

Objetos gráficos

El motor gráfico de IDL es orientado a objetos. Una jerarquía de objetos que provee funcionalidades gráficas independientes de la plataforma de uso o los dispositivos de salida.

Ofrece una serie de objetos a partir de los cuales se puede construir la arquitectura de la aplicación. Dicha estructura, de características jerárquicas, se basa en una ventana (`IDLgrWindow`), en la que se agregan vistas (`IDLgrView`), compuestas de modelos (`IDLgrModel`). Los modelos son las entidades a dibujar. Están compuestos por objetos atómicos (líneas, polígonos, etc) o por otros modelos. Estos objetos atómicos contienen los puntos y las conectividades a dibujar -además de los atributos gráficos-; los *modelos* tienen los atributos y las capacidades que permiten las rotaciones y traslaciones de los objetos, dentro de la ventana de dibujo. Esta clase maneja una matriz de rotación y traslación (de 4x4), y de escala, que se abstraen al usuario a través de funciones como `Translate`, `Rotate` y `Scale`.

De esta forma, se dibujan estructuras muy complejas, con grandes cantidades de datos, de forma muy eficiente y simple. Resolviendo cuestiones de graficación que no son el centro del problema que se quiere resolver. Por ejemplo, un conjunto de líneas -`IDLgrPolylines`- y cierta conectividad definen una grilla -`IDLgrModel`- que las agrupa y las maneja como un ente atómico que se puede mover, ocultar, borrar, etc; tratándolas como un conjunto, liberando al programador de tener que preocuparse por cada objeto

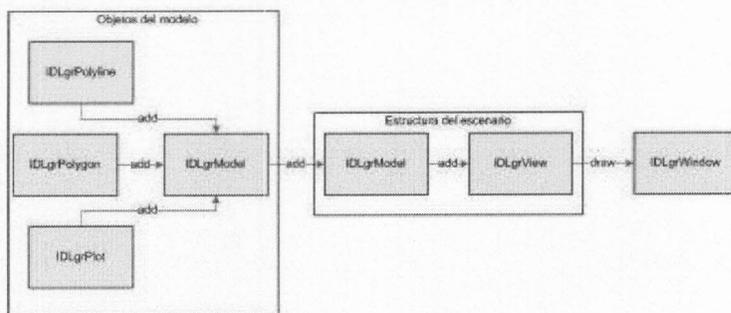


Figura 8.1: Jerarquía de objetos gráficos de IDL

que lo compone.

8.2. Implementación de GEOSAR: esquema general

La aplicación fue desarrollada bajo el paradigma orientado a objetos. Se diseñó una arquitectura teniendo en cuenta las características de IDL para el manejo de objetos gráficos; además de las de la relación de los objetos del problema y los requerimientos.

8.2.1. El propagador

Es necesario que se mantenga cierta independencia entre el propagador implementado y el resto del programa, para que sea fácil de cambiar por cualquier otro. Además, la necesidad de exportar las funciones de propagación a otros módulos dentro del proyecto, es otra razón para tal separación. Al programar bajo el paradigma orientado a objetos, esta independencia, se obtiene de forma directa al declarar la interfaz de la clase y esconder la implementación.

Entonces, para aislar el módulo de propagación, fue definida una función, que hace de interfaz. Se especificó el nombre y los parámetros que pueden implementar los distintos modelos, sin afectar el resto de la aplicación.

La función se define como se muestra a continuación:

```

PROPAGAR( IN tiempos:[JULIANO],
          OUT ECI_POS:[x,y,z],
          OUT ECI_VEL:[x,y,z])
  
```

Donde `tiempos` es un vector de fechas en formato juliano; y `ECI_POS` y `ECI_VEL` son los vectores de posición y velocidad devueltos por el propagador.

La programación en IDL debe maximizar el uso de las rutinas optimizadas para el cálculo vectorial, y minimizar la utilización de ciclos. Por esta razón, la implementación del propagador fue modificada levemente para que acepte como parámetro un vector de fechas, en lugar de un solo tiempo.

8.2.2. La escena

La aplicación principal se pensó como un entorno compuesto de un mapa, sobre el cual se agregan otros objetos gráficos.

Es decir que hay una entidad principal -clase `Escena`-, a la que se agregan el resto de los objetos.

Esta entidad mantiene actualizado y consistente todo el entorno gráfico. En particular, mantiene la consistencia entre la proyección de las coordenadas de latitud y longitud que utilizan los distintos objetos.

El tipo de proyección y el centro del mapa son propiedades configurables. Al cambiarse, deben recalcularse las coordenadas cartesianas de todos los objetos del mapa, y actualizarse según corresponda.

Cada objeto real, se caracteriza por su posición en coordenadas geográficas. Para dibujarlos, se obtienen los puntos en coordenadas cartesianas según la proyección de mapa que se esté usando en ese momento, se carga la información en los objetos gráficos y se dibuja la ventana. Cuando se cambia la proyección, hay que recalcular las posiciones en coordenadas cartesianas y redibujar cada objeto en la escena. Este procedimiento, además de ser muy costoso, debe asegurar la coherencia en la proyección que usa cada objeto. Para resolver este problema se optó por un esquema donde la proyección sea un objeto único administrado por la `Escena`. Cuando se produzca un cambio como el descrito antes, la escena ordenará a cada objeto, a que se redibuje, pasándole como parámetro la nueva proyección.

Pero éste es sólo un aspecto de esta interacción entre la entidad coordinadora y los suscriptos. El otro lado -la actualización de los objetos particulares- se describe más adelante.

8.2.3. Los objetos gráficos

Cada objeto que se ve en la pantalla de GEOSAR representa una entidad en el mundo real y se pretende que cada una sea lo más independiente posible. Guardará sus coordenadas geográficas de forma conveniente y se usarán

a la hora de dibujarlo, como se describe en el párrafo acerca del *ObjGráfico*. Claro que como el conjunto debe ser coherente entre sí, la independencia no será total.

Todos heredan de otros dos objetos principales: de *IDLgrModel*, por lo explicado acerca del entorno de programación de IDL, para encapsular en el una estructura más compleja, y de *ObjGráfico* porque define la interfaz para incorporarse al esquema de la aplicación.

El objeto *ObjGráfico*

Esta interfaz integra los objetos en la Escena y define el mecanismo de comunicación y actualización. Cuenta con tres funciones que llevan a cabo este trabajo:

- **Adjuntar**: este método es el primero que aparece en el protocolo de comunicación entre los objetos gráficos y el contenedor del escenario. Recibe como parámetro la *Escena*, que luego se utiliza para obtener el objeto *Proyección*.
- **Actualizar**: que no recibe parámetros, se encarga de hacer la actualización del objeto cuando se haya producido un cambio que signifique redibujarlo completamente. *Actualizar* obtiene el objeto que proyecta las coordenadas geográficas en coordenadas cartesianas, y se lo pasa a *Dibujar*.
- **Dibujar**: es un método virtual. Se implementa en las clases que heredan de *ObjGráfico*. La idea es que esta función haga las tareas necesarias para generar el objeto gráfico.

Definida esta interfaz, agregar nuevos objetos a la escena no es complicado, solo deben heredar de *ObjGráfico* e implementar el método *Dibujar*.

8.2.4. Los layers

Como se explicó en la sección 7.4.4 la aplicación tiene un mecanismo para agregar información en máscaras que se agregan a la *Escena*.

En el contexto de esta sección, el mecanismo de *Layers* permite agrupar objetos gráficos. Es necesario hacer una diferenciación con el concepto de *Layer de información* en el contexto de un GIS. Cada capa de información que se agrega al programa, genera un *Layer* nuevo. Cada objeto que se genere

con GEOSAR se agrega a un Layer, que se puede exportar en una capa de información.

En definitiva, lo que se describe en esta sección, es el mecanismo para soportar la funcionalidad para incluir capas de información externa.

El objeto IDLgrModel, ya mencionado, es apropiado para modelar estas capas.

Hay una clase que las maneja de forma adecuada. Los objetos gráficos sólo ven la escena, a ella se adjuntan, y es ella quien se encarga de reenviar los mensajes a la clase que administra los layers. Dichos mensajes dependen de la clase que se esté agregando, así por ejemplo, las capas nuevas se agregan en una pila, y los objetos gráficos se agregan a la capa activa.

8.2.5. Esquema de graficación

Por último, y para una mejor comprensión de la sección, se muestra en la figura 8.2 un esquema con la dinámica de los objetos que se nombraron en los apartados anteriores. Se detalla como interactúan y cual es el vinculo con los distintos objetos.

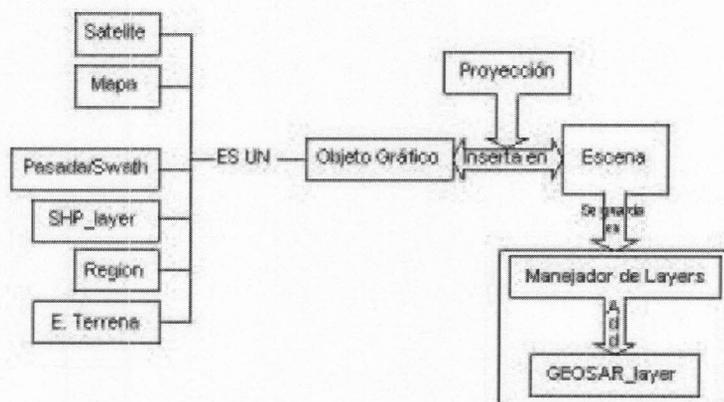


Figura 8.2: Esquema de los objetos de la interfaz grafica

8.3. Comunicación con el planificador

La comunicación con el planificador se hace por medio de intercambio de archivos. Este esquema, que en principio puede parecer lento, resulta apropiado para el volumen de información que se intercambia -que en general no supera el kilobyte- y la frecuencia, no intensiva, de las interacciones. En contraposición, el intercambio de archivos maximiza la independencia entre las aplicaciones que, al momento de la planificación, siguen pareciendo indistinguibles para el usuario.

La sincronización se realiza por medio de un mecanismo de semáforo. Impulsados por el evento de un *timer* ambos programas consultan un directorio en común hasta encontrar un archivo que indica la existencia de información para procesar. En ese momento se lee un archivo con los datos y se actúa en consecuencia. La acción a llevar a cabo está determinada por el nombre del archivo según se detalla en las tablas siguientes:

Nombre	Origen	Significado
orbita_ull.dat	Planificador	Cálculo de ángulos para un punto en el terreno.
orbita_ull.flg	Planificador	Mensaje de datos de calculo de ángulos para leer
orbita_swt.dat	Planificador	Datos para el dibujo de swath
orbita_swt.flg	Planificador	Mensaje de datos de swath para leer
orbita_ran.dat	Planificador	Cálculo de distancia en rango.
orbita_ran.flg	Planificador	Mensaje de datos de calculo de rangos para leer
orbita.ocp	Planificador	El planificador no puede recibir datos

Cuadro 8.1: Archivos de comunicación planificador-GEOSAR

El protocolo, que parece demasiado sencillo, sirve en el contexto de estas interacciones, que también son muy sencillas. La secuencia siempre la inicia el mismo programa (el planificador), y, en el caso de la consulta de ángulos de visibilidad o rangos, espera que finalice el cálculo para seguir procesando. En

Nombre	Origen	Significado
plansar_ull.dat	GEOSAR	Cálculo de ángulos para un punto en el terreno.
plansar_ull.flg	GEOSAR	Mensaje de datos de calculo de ángulos para leer
plansar_swt.dat	GEOSAR	Datos para el dibujo de swath
plnsar_swt.flg	GEOSAR	Mensaje de datos de swath para leer
plansar_ran.dat	GEOSAR	Cálculo de distancia en rango.
plansar_ran.flg	GEOSAR	Mensaje de datos de calculo de rangos para leer
plansar.ocp	GEOSAR	El planificador no puede recibir datos

Cuadro 8.2: Archivos de comunicación GEOSAR-planificador

cambio, cuando el pedido es para graficar un swath, luego de enviar el mensaje no necesita ningún otro de respuesta por lo que no espera y sigue con la planificación. Además, aunque al momento de graficar, el usuario desviará su atención a la ventana de GEOSAR, existe otra señal de *Ocupado* por la que ambos programas, al leerla, comprenden que el otro lado esta procesando y no es posible enviarle mensajes.

A continuación, se exponen los esquemas de interacción con el planificador. En ellos se muestran los componentes del módulo de comunicación, y la forma en que interactúan.

El primero (figura 8.3) es una descripción del módulo y los vínculos entre los objetos que lo forman o interactúan para obtener resultados.

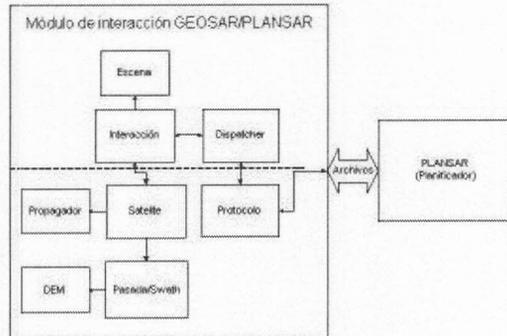


Figura 8.3: Objetos utilizados en el módulo comunicación

8.3.1. Esquema de interacción para calcular rangos

Para el cálculo de rangos y ángulos, la dinámica de los objetos es parecida. En la figura 8.4 solo se muestra el primer caso. Los números de las etiquetas en los vínculos, indican el orden en que se da cada uno de los eventos.

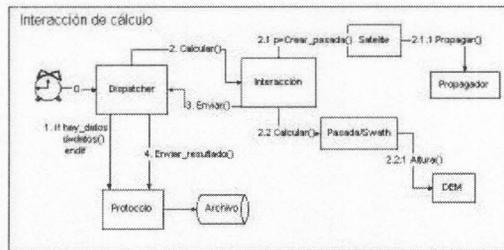


Figura 8.4: Dinámica de los objetos en la interacción de cálculo de rangos

8.3.2. Esquema interacción para graficar una pasada

Para graficar las pasadas por orden del planificador, se realizan los pasos que se describen en la figura 8.5. Los números de las etiquetas en los vínculos, indican el orden en que se da cada uno de los eventos.

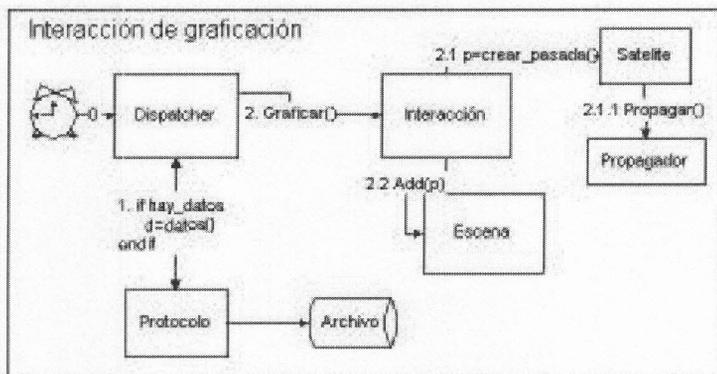


Figura 8.5: Dinámica de los objetos en la interacción para graficar una pasada

8.3.3. Extensibilidad

Teniendo en cuenta que la aplicación se pensó para hacer posibles futuras extensiones y modificaciones, el mecanismo de comunicación no podía pensarse de otra manera. Si bien el que se muestra aquí resulta a medida de las necesidades del Planificador, agregar nuevas consultas solo implica agregar nuevos archivos de comando y programar su interpretación. Es decir, trabajo para la extensión; pero no una modificación sustancial del programa. Lo mismo si se coloca otro programa en lugar del Planificador. El GEOSAR responderá de la misma manera para cualquier software que genere consultas de forma correcta.

Otro cambio a tener en cuenta, es el del medio de comunicación. Si en lugar de un archivo y un directorio, se colocara una red, el GEOSAR debería adaptarse sin mayores esfuerzos, debido a que la escritura en el medio se encapsula en la clase *Protocolo*, que exporta el método *ReadData* y *WriteData*.

De todos modos, no debe perderse de vista que esta implementación es para una interacción sencilla. Agregar complejidad podría obligar a implementar mecanismos que se dejaron afuera, como la comunicación de errores, y la verificación de la correctitud de los datos.

8.4. Representación del tiempo

La forma de representación del tiempo, se refiere al modo en que se almacenan las fechas y a su interpretación. En la aplicación estas variables

aparecen continuamente pues las funcionalidades principales tienen ventanas de tiempo como inputs.

En los capítulos de test del propagador se comentó la importancia de manejar los tiempos con buena precisión. Por otro lado, en varios cálculos se hacen cuentas en las que se opera con fechas. Todo esto es mucho más simple si, en lugar de utilizar fechas al estilo del calendario gregoriano (día/mes/año, etc), se utiliza un valor que sea la cantidad de días transcurridos desde cierta fecha.

La suma de estos aspectos resulta en la forma que se adoptó para representar el tiempo internamente: valores que representan una cantidad de días desde una fecha de referencia. Y en particular, se implementaron dos posibilidades, días julianos (JD) y días julianos modificados MJD

Estas dos formas que se utilizan difieren en el momento en que empiezan a contar los días. El por qué de la decisión queda claro en el capítulo de test del propagador: hay que manejar números con buena precisión. Por eso la fecha desde las cuales comienzan a contar los días, dependerá de los requerimientos de precisión. Si, por ejemplo, se necesitara propagar con diferencias del segundo se necesitarán, al menos, 5 decimales (1 día tiene aproximadamente 86400 segundos). Una fecha cercana, en julianos, utiliza 7 dígitos para la cantidad de días (el día juliano correspondiente al 1/11/2004 a las 12:00:00 es el 2453311), por lo que se necesitarán, al menos, 12 dígitos significativos en la representación. IDL maneja 5 dígitos decimales para representar fechas en las funciones JULDAY y CALDAT.

Si se quiere propagar con diferencias menores al segundo, entonces será necesario cambiar la representación. Los días julianos modificados se definen como $MJD = JD - 2400000,5$ lo que da dos dígitos más para diferencias más pequeñas.

Sin embargo, se optó por mantener ambas representaciones, debido a que, ciertas operaciones (calcular el ángulo entre el meridiano principal y el equinoccio de una fecha, para la conversión ECI-ECEF) necesitan de la representación en días julianos tradicionales.

Además, en el capítulo de test del propagador, se utilizó como fecha de referencia, el 1/1/1950, para minimizar el error de los JD y los MJD (todavía muy grandes) y obtener resultados del error del propagador, y no de las representaciones de la fecha.

Para la interfaz con el usuario, se incluyen funciones de conversión entre las distintas formas de representación (MJD, JD, gregoriano) para mostrar las fechas de la mejor forma, según el contexto.

Capítulo 9

Conclusiones

Como conclusión del trabajo se puede decir que el software creado cumple con las expectativas de cuando fue pensado. Que es un producto de software que integra un modelo de propagación complejo, en un ambiente gráfico para un usuario final no experto. Es decir, que aunque las bases -los fundamentos teóricos- de las funcionalidades son complejas, las facilidades gráficas hacen que el programa pueda ser utilizado por alguien que no esté al tanto de los detalles de tales fundamentos. Que solo necesite propagar la órbita en función del tiempo.

Como asistente del planificador, brinda los servicios que se le solicitan, pero a la vez permite que sean útiles para otros módulos al ser de fácil adaptación, al tener una interfaz simple que lo comunica con el exterior. Estas ventajas se podrán verificar con la creación de nuevos paquetes de software que se vayan sumando al conjunto del software de vuelo.

Incluso se podrían explotar sus características como GIS, o agregarle nuevas funcionalidades, para lograr una herramienta con múltiples propósitos, más allá de la *aplicación para satélites*. Cualquier información que sea útil desplegar en un mapa puede ser incluida en el programa. En ese sentido, se puede decir que existen algunos trabajos inmediatos, que pueden agregarse al software, para darle más fuerza.

Algunas extensiones para la versión actual serí:

- Es directo (y está implementado en versiones siguientes) el paso a *n* satélites.
- En versiones posteriores también se incluyen estaciones terrenas, con

cálculos de visibilidad de la estación (link) y tiempos de pasada.

- Ampliar el espectro de formatos de GIS que maneje (dxf).
- Aumentar sus capacidades para graficar información georreferenciada en formatos RASTER.
- Inclusión de otros métodos de propagación, o inputs para definir la órbita.

Completando estos trabajos finales, ya estaríamos en condiciones de hablar de una herramienta muy completa, e ideal para la edición, en general, de escenarios de misión.

Apéndice A

Función de fuerza

En un sistema de n masas con coordenadas x_i, y_i, z_i se define el potencial en el punto m_i como

$$V_i = G \sum_{j=1}^n \frac{m_j}{r_{ij}} \quad (j \neq i), \quad (\text{A.1})$$

De la segunda ley de Newton y de la ley de gravitación universal surge que

$$m_i \frac{d^2 x_i}{dt^2} = G m_i \frac{\partial}{\partial x_i} \sum_{j=1}^n \frac{m_j}{r_{ij}} \quad (j \neq i), \quad (\text{A.2})$$

donde el miembro derecho de la igualdad representa la fuerza ejercida sobre la i -ésima partícula y el izquierdo el producto de la masa de dicha partícula por su aceleración en la dirección x . Entonces, de (A.1) y (A.2) puede verse que

$$m_i \frac{d^2 x_i}{dt^2} = m_i \frac{\partial V_i}{\partial x_i}. \quad (\text{A.3})$$

Si definimos la magnitud U , a la que llamaremos “función de fuerza” y que posee dimensiones de energía, como

$$U = \sum_{i=1}^{n-1} \sum_{j=2}^n \frac{G m_i m_j}{r_{ij}} \quad (j \neq i),$$

vemos que

$$\frac{\partial U}{\partial x_i} = m_i \frac{\partial V_i}{\partial x_i}, \quad (\text{A.4})$$

por lo que reescribiendo (A.2) utilizando (A.3) y (A.4) para las tres direcciones del espacio tenemos

$$m_i \frac{d^2 x_i}{dt^2} = \frac{\partial U}{\partial x_i} \quad (\text{A.5})$$

$$m_i \frac{d^2 y_i}{dt^2} = \frac{\partial U}{\partial y_i} \quad (\text{A.6})$$

$$m_i \frac{d^2 z_i}{dt^2} = \frac{\partial U}{\partial z_i}. \quad (\text{A.7})$$

Por lo tanto, el conocimiento de U es útil en la determinación del movimiento cada una de las n masas del sistema.

Apéndice B

Gráficos del ejemplo de uso

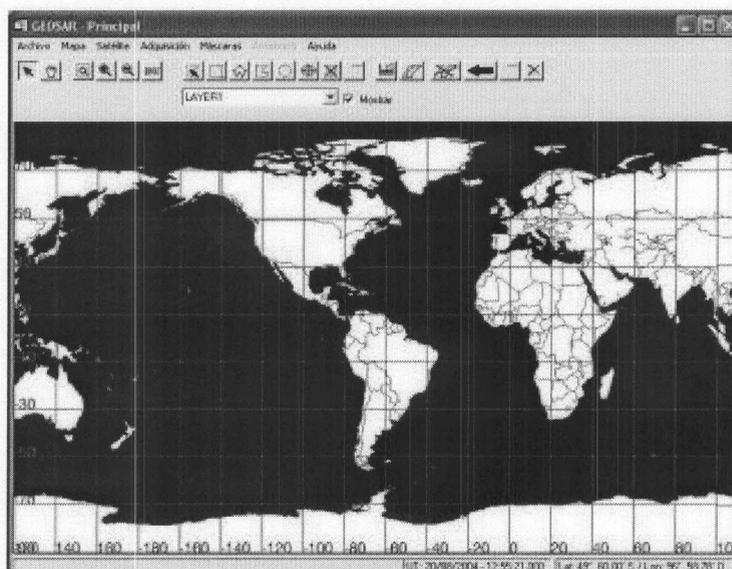


Figura B.1: La aplicación en su estado inicial

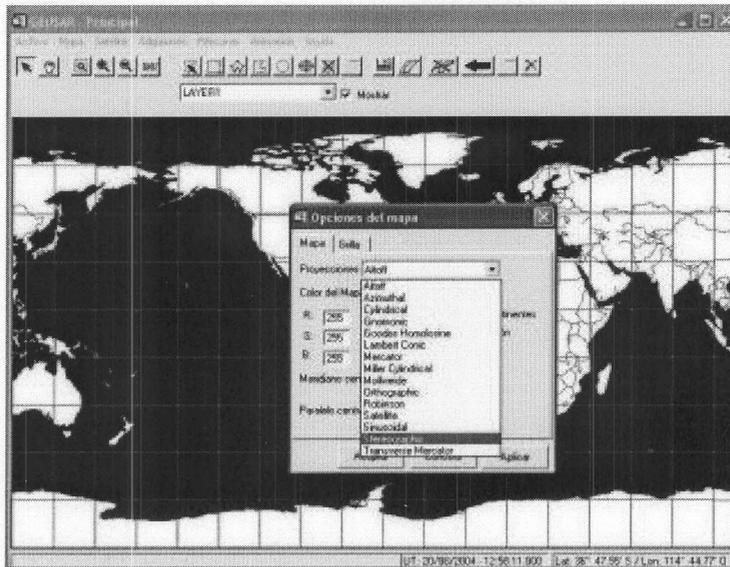


Figura B.2: Ventana para elegir la proyección

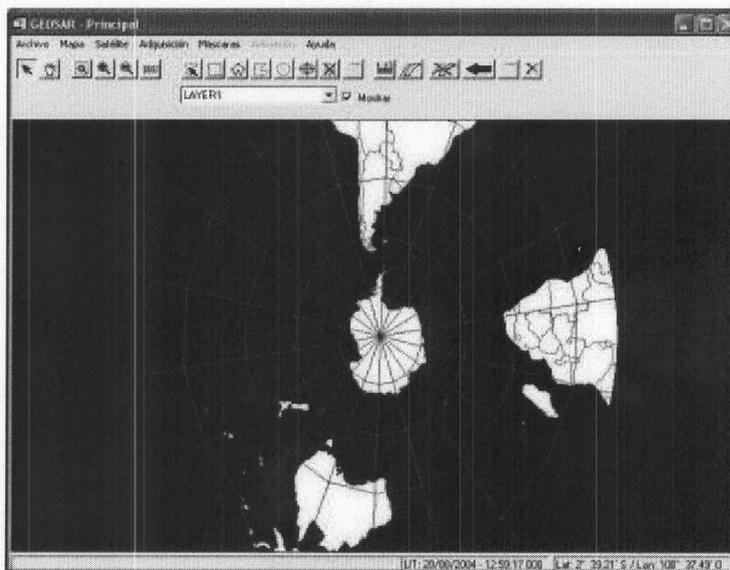


Figura B.3: Planisferio luego de aplicar la proyección

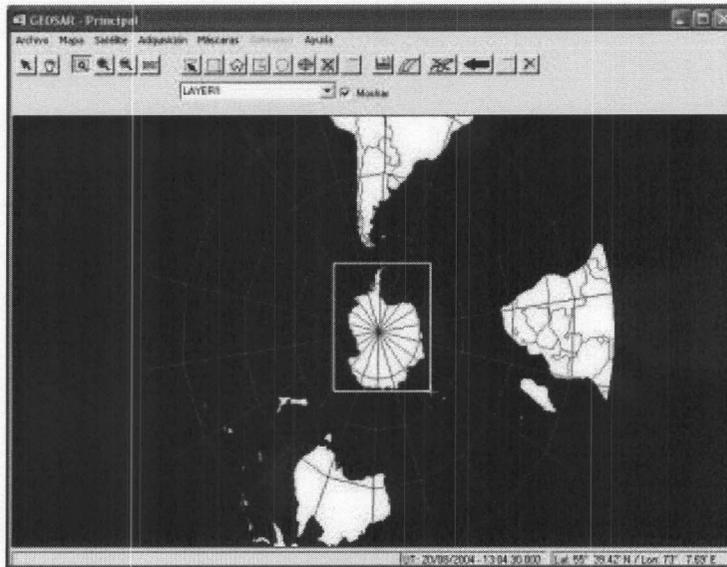


Figura B.4: Herramienta para manejo de zoom en el mapa

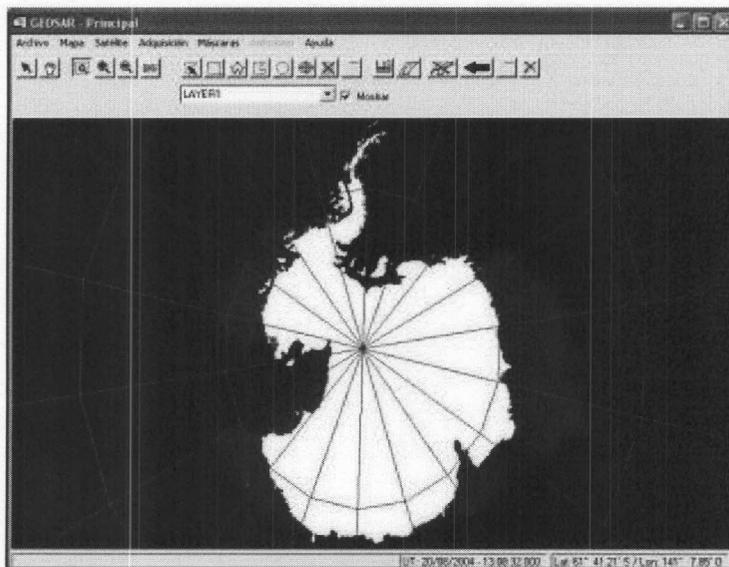


Figura B.5: Acercamiento a la antártida

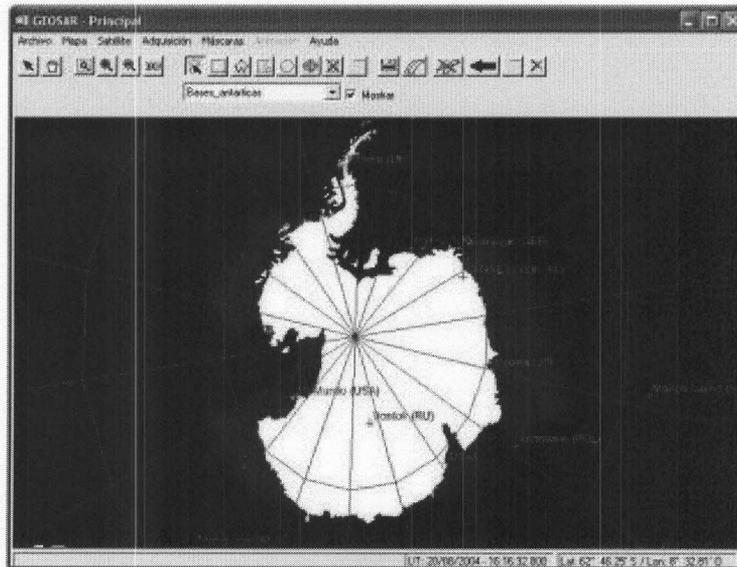


Figura B.6: Capa de información: bases sub-antárticas

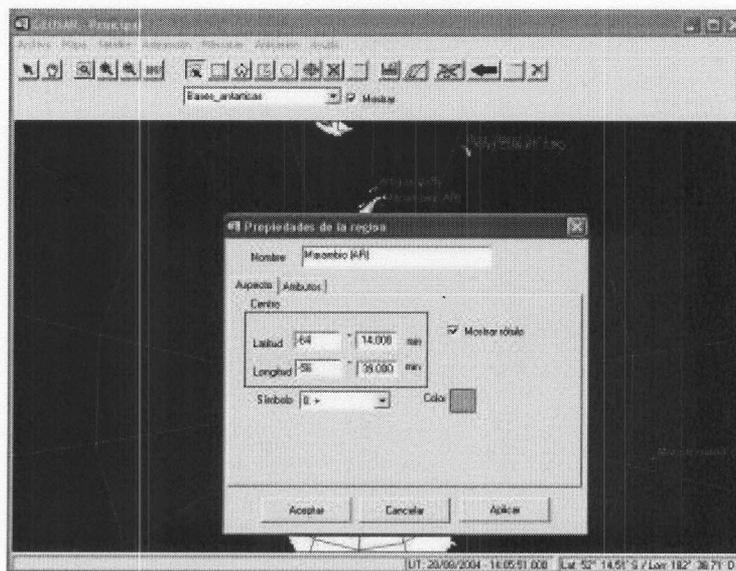


Figura B.7: Ventana para cálculo de ángulos

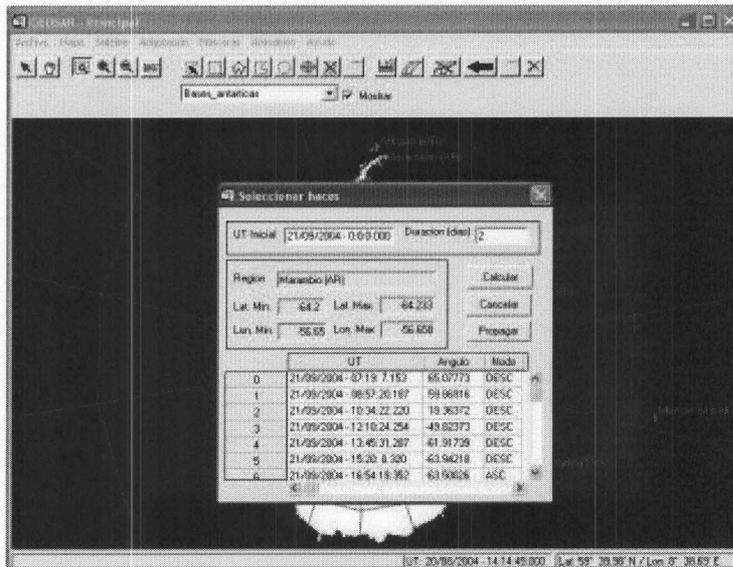


Figura B.8: Ángulos obtenidos del cálculo

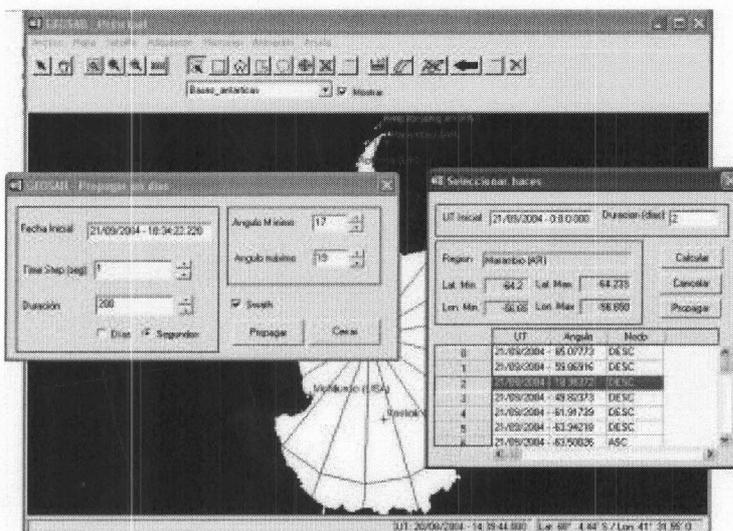


Figura B.9: Ventanas de cálculo de ángulos y dibujo del swath

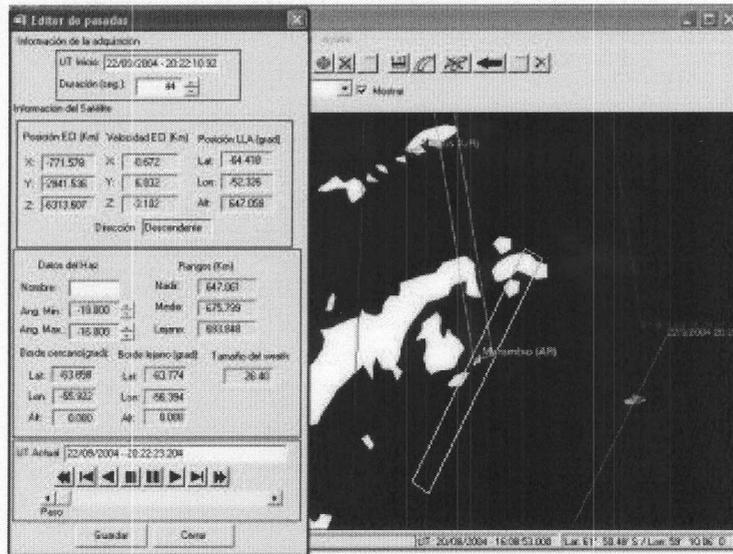


Figura B.10: Ventana de edición de propiedades del swath

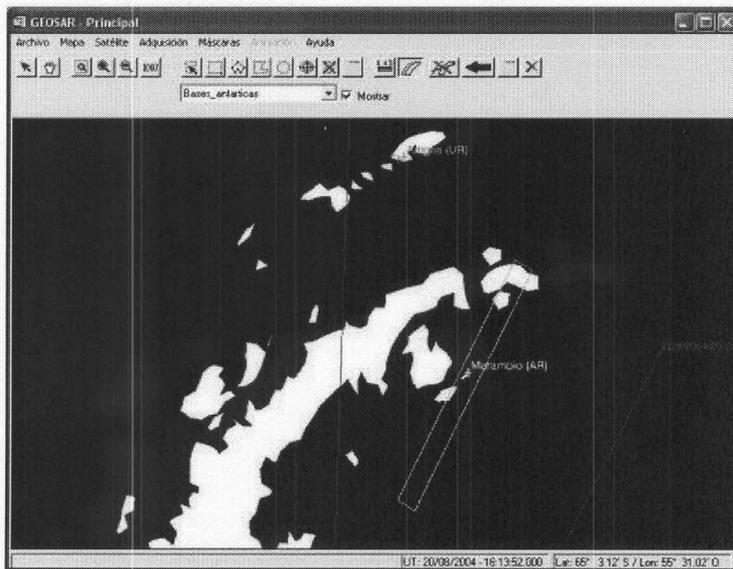


Figura B.11: Swath elegido

Bibliografía

- [Bro59] D. Brouwer. Solution of the problem of artificial satellite theory without drag. *Astronomical Journal*, 64:378–+, November 1959.
- [FRH88] R. L. Roehrich F. R. Hoots. *Spacetrack Report Nr. 3. Models for Propagation of NORAD Elements Sets*. 1988.
- [JCC91] R.Ñ. McDonough J. C. Curlander. *Synthetic Aperture Radar Systems and Signal Processing*. Wiley-Interscience, 1991.
- [Koz59] Y. Kozai. The motion of a close earth satellite. *Astronomical Journal*, 64:367–+, nov 1959.
- [NAS89] NASA. *Goddard Trajectory Determination System (GTDS). Mathematical Theory, Revision 1, Chap. 3*. 1989.
- [Sei92] P. K. Seidelmann. *Explanatory Supplement to the Astronomical Almanac*. 1992.
- [Sys99] Research Systems. *Using IDL*, 1999.
- [USG88] USGS. Gtopo30 documentation. 1988.
- [USG98] USGS. Wgs84 implementation manual. 1998.
- [Zad98] P. E. Zadunaisky. *Introducción a la astrodinámica*. Comisión Nacional de Actividades Espaciales (CONAE), 1998.