

Trabajo de Cátedra

Calculadora Geoespacial

Presentación

EST 2012 – Concurso de Trabajos Estudiantiles

JAIIO 41

Facultad de Informática de la Universidad de la Plata

2012

Autor

Ariel Ivan Diaz

arielivandiaz@ieee.org

Cátedra

Informática I

Primer Año Ingeniería Electrónica

Profesor: Ricardo Coppo

ricardo.coppo@gmail.com

Universidad Tecnológica Nacional Facultad Regional Bahía Blanca



Introducción a los cálculos geospaciales en C

1 Resumen

El objetivo de este proyecto es crear un programa que realice cálculos geospaciales, usando los contenidos aprendidos en la cátedra de Informática I, de primer año, Ingeniería Electrónica. Dada la naturaleza del proyecto se debió realizar una investigación sobre los conceptos geográficos, geodésicos y matemáticos para los cálculos astronómicos implementados.

La matemática utilizada se divide en dos áreas: los sistemas de representación y el cálculo de la distancia entre dos puntos. Se aproximó la geometría de la Tierra usando dos modelos: la Esfera y Elipsoide Terrestre. En el primer caso la distancia se calcula usando la fórmula de haversine^{[1][4]} y en el segundo caso se utilizó la fórmula de Vincenty^{[2][3]}. Una vez analizados los conceptos necesarios se procedió a desarrollar el programa “Calculadora Geoespacial”, e integrar las distintas fórmulas matemáticas como algoritmos de programación. Finalmente se realizaron pruebas de comparación con los resultados obtenidos.

Palabras clave: Geodesia, Coordenadas Geográficas, Conversión de Coordenadas, Cálculo de Distancia, Haversine, Vincenty.

2 Introducción

Se buscó encontrar una aplicación práctica para aplicar los conceptos de programación, para eso se creó el programa “Calculadora Geoespacial”. El objetivo del programa es que en base a coordenadas geográficas pueda realizar operaciones matemáticas con un propósito determinado. El programa se basa en los distintos métodos de cálculo geospaciales, los cuales debieron ser desarrollados e implementados en el lenguaje de programación. Para utilizar estos métodos se trabajó sobre el procesamiento de los datos para adaptarlos a las distintas funciones. A lo largo de todo el programa, se buscó sencillez y robustez, para poder facilitar su adaptación y mejora, dado que si bien en este proyecto no se tuvo en cuenta una aplicación práctica inmediata se consideró esta posibilidad a futuro.

Dentro del programa se consideró la conversión de estas coordenadas geográficas a coordenadas cartesianas, dado que estas últimas son más fáciles de visualizar y operar. En los casos del cálculo de distancia entre dos puntos, se usaron valores angulares derivados directamente de las coordenadas geográficas para ser

aplicados en las fórmulas de haversine y Vincenty. En los diferentes casos se aproxima la geometría de la Tierra, ya sea como Esfera o como Elipsoide, dadas estas aproximaciones se creyó conveniente el poder comparar los valores de los distintos modelos y la diferencia de estos.

3 Conceptos matemáticos

3.1 Sistemas de representación

La representación geométrica exacta de la Tierra no es posible, dado la complejidad del cuerpo. Es por esto que en la mayoría de los casos se usan modelos simplificados para este fin. Para este trabajo utilizaremos en primer medida el modelo de Esfera Terrestre y para una aproximación más cercana el modelo de Elipsoide Terrestre. Un modelo más complejo aun es el de Geoide, este solido toma como nivel medio al Elipsoide Terrestre y aproxima sobre este la superficie topográfica. El geoide es muy difícil de representar geoméricamente y por ello que no fue usado en este trabajo.

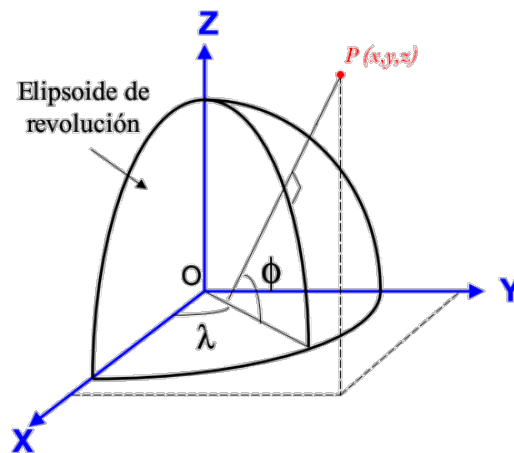


Figura 1

3.1.1 Coordenadas geográficas

Los sistemas de coordenadas geográficas son las más utilizado para definir un punto sobre la superficie terrestre. Una localización en este tipo de sistema queda definido por dos valores angulares: Latitud (ϕ) y Longitud (λ).

3.1.2 Coordenadas cartesianas

Para la realización de algunos cálculos es necesario transformar las coordenadas geográficas en coordenadas cartesianas ortonormales en el espacio (x,y,z) con origen en el centro de la Tierra.

3.1.2.1 Representación esférica

Para esta representación se tiene como dato que $6371 \text{ km}^{[5]}$ es el radio propuesto para la Tierra.

$$\begin{cases} x = 6371 \cdot \cos(\phi) \cdot \cos(\lambda) \\ y = 6371 \cdot \cos(\phi) \cdot \text{sen}(\lambda) \\ z = 6371 \cdot \text{sen}(\phi) \end{cases}$$

ϕ = Valor que resulta de la Latitud

λ = Valor que resulta de la Longitud

3.1.2.2 Representación elipsoidal

La representación en este caso está dada por:

$$\begin{cases} x = 6378,14 \cdot \cos(\phi) \cdot \cos(\lambda) \\ y = 6378,14 \cdot \cos(\phi) \cdot \text{sen}(\lambda) \\ z = 6356,755288 \cdot \text{sen}(\phi) \end{cases}$$

Donde $6378,14 \text{ Km}$ es el radio mayor medido en sobre el Ecuador y $6356,755288 \text{ Km}$ es el radio menor dado por el achatamiento $1/298,257$. Estos valores de constantes astronómicas fueron adoptados en la Asamblea General de la Unión Astronómica Internacional que se celebró en Grenoble en 1976, designado por IAU (1976), que entró en vigor el 1 de enero de 1984. Con estos valores obtenemos al elipsoide denominado WGS 84^{[5][6]}.

3.2 Fórmulas de distancia

3.2.1 Distancia Euclidiana

En el caso de la geometría de la Tierra, la ecuación Euclidiana de la distancia no resulta adecuada, debido que la medida no se toma sobre la superficie. Esta resulta una buena aproximación para distancias en las cuales la curvatura de la Tierra puede ser despreciable.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

3.2.2 Fórmula de haversine

La fórmula de haversine^{[1][4]} es un caso especial de la ley de haversines, una fórmula general de la trigonometría esférica muy utilizada en la navegación. La función se expresa en términos de funciones de haversine o múltiplos de esta. Se define como:

$$\text{haversine}\left(\frac{D}{R}\right) = \text{haversine}(\phi_1 - \phi_2) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \text{haversine}(\Delta\lambda)$$

Dónde:

$$\text{haversine}(\theta) = \sin^2\left(\frac{\theta}{2}\right)$$

D = distancia entre los 2 puntos

R = radio de la esfera

ϕ = latitud

λ = longitudud

$\Delta\phi$ = diferencia de latitud

$\Delta\lambda$ = diferencia de longitudud

Al despejar la distancia de la fórmula de haversine se obtiene una nueva ecuación, que presenta algunos problemas para ciertos hemisferios, para solucionar estos se propone la fórmula directa de la distancia de haversine:

$$\Delta\phi = \phi_1 - \phi_2$$

$$\Delta\lambda = \lambda_2 - \lambda_1$$

$$A = \sin^2\left(\frac{\Delta\phi}{2}\right) \cdot \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)$$

$$C = 2 \cdot \text{atan2}(\sqrt{A}, \sqrt{1-A})$$

$$D = R \cdot C$$

Esta nueva función presenta problemas cuando se trata de puntos antípodas o cercanos a serlo.

3.2.3 Fórmula de Vincenty

Este método iterativo fue publicado por Thaddeus Vincenty en 1975^[3] y constituye una serie de fórmulas que forman un algoritmo muy eficiente para el cálculo de distancia sobre el modelo de elipsoide terrestre, lo cual hace que sea más precisa y real la medición. A continuación se presenta la fórmula de Vincenty.

Sean:

a, b = semiejes mayor y menor del elipsoide

ϕ_1, ϕ_2 = latitud geodésica

$$\begin{aligned}
L &= \text{diferencia de longitud} \\
f &= \text{achatamiento del elipsoide} \\
U1 &= \text{atan}((1 - f) - \tan(\phi_1)) \\
U2 &= \text{atan}((1 - f) \cdot \tan(\phi_2)) \\
\lambda &= L \\
\lambda' &= 2\pi
\end{aligned}$$

Entonces la distancia (s) entre dos puntos se calcula usando:

$$\begin{aligned}
&\text{do while}(\lambda - \lambda') > 10^{-12} \quad (\text{implica un error} < 0.06\text{mm}) \\
\{ & \\
&\sin(\sigma) = \sqrt{(\cos(U2) \cdot \sin(\lambda))^2 + (\cos(U1) \cdot \sin(U2) - \sin(U1) \cdot \cos(U2) \cdot \cos(\lambda))^2} \\
&\cos(\sigma) = \sin(U1) \cdot \sin(U2) + \cos(U1) \cdot \cos(U2) \cdot \cos(\lambda) \\
&\sigma = \text{atan2}(\sin(\sigma), \cos(\sigma)) \\
&\sin(\alpha) = \cos(U1) \cdot \cos(U2) \cdot \frac{\sin(\lambda)}{\sin(\sigma)} \\
&\cos^2(\alpha) = 1 - \sin^2(\alpha) \\
&\cos(2\sigma m) = \cos(\sigma) - \frac{2 \sin(U1) \cdot \sin(U2)}{\cos^2(\alpha)} \\
&C = \frac{f}{16} \cos^2(\alpha) \cdot [4 + f \cdot (4 - 3\cos^2(\alpha))] \\
&\lambda' = \lambda \\
&\lambda' = L + (1 - C)f \cdot \sin(\alpha) \{ \sigma + C \cdot \sin(\sigma) [\cos(2\sigma m) + C \cdot \cos(\sigma) (-1 + 2\cos^2(2\sigma m))] \} \\
&\quad (\text{útese } \cos(2\sigma m) = 0 \text{ si se está a lo largo del ecuador}) \\
&\}
\end{aligned}$$

$$\begin{aligned}
u^2 &= \cos^2(\alpha) \frac{a^2 - b^2}{b^2} \\
A &= 1 + \frac{u^2}{16384} \{ 4096 + u^2 \cdot [-768 + u^2(320 - 175u^2)] \} \\
B &= \frac{u^2}{1024} \cdot \{ 256 + u^2 \cdot [-128 + u^2 \cdot (74 - 47u^2)] \} \\
\Delta\sigma &= B \sin(\sigma) \cdot \{ \cos(2\sigma m) + \frac{B}{4} \cdot [\cos(\sigma) \cdot (-1 + 2\cos^2(2\sigma m)) \\
&\quad - \frac{B}{6} \cdot \cos(2\sigma m) \cdot (-3 + 4\sin^2(\sigma)) \cdot (-3 + 4\cos^2(2\sigma m))] \} \\
s &= b \cdot A \cdot (\sigma - \Delta\sigma)
\end{aligned}$$

Al igual que en el método anterior, esta función también presenta problemas en puntos antípodas.

4 Desarrollo del programa

4.1 Estructura del Programa

Se organizó la estructura en 3 archivos, para facilitar su comprensión, flexibilidad y actualización. Un primer archivo contiene las variables matemáticas y astronómicas usadas, así como también define los tipos de estructuras usadas para almacenar los distintos datos de entrada y salida. En un segundo archivo se encuentran todas las funciones creadas y se incluyen las distintas bibliotecas que serán utilizadas. El último archivo es el principal y es el que vincula a los dos anteriores. En este archivo se encuentra el programa principal, el cual es responsable de invocar a las funciones y la secuencia en que se ejecutan.

En el módulo donde se definen las variables globales, se declaran las constantes astronómicas que luego serán usadas en el programa, como es el caso del radio de la esfera^[5] terrestre y los radios del elipsoide^{[5][6]} terrestre. También se definen nuevos tipos de estructuras, las cuales sirven para almacenar la información de un punto en las distintas instancias del programa.

Las totalidades de las funciones están declaradas en un segundo módulo, en el cual también se incluyen las bibliotecas propias del lenguaje a utilizar. En este archivo se encuentran las funciones de entrada y salida de datos, las funciones que manejan los archivos de historial y de datos, y las funciones matemáticas. Las principales funciones de cálculo matemático resultan ser análogas a sus respectivas formulas anteriormente mencionadas en la sección de conceptos matemáticos.

En el módulo principal, se enlazan los módulos anteriores y se maneja la secuencia en la cual se presenta y ejecuta el programa. Este resulta ser muy sencillo dado que solo cita a la función que el usuario llama y esta misma se encarga de la entrada, procesamiento y salida de datos.

El programa crea un archivo de texto en el cual registra los resultados de cada ejecución. Este archivo toma su nombre de la hora del sistema en la cual fue ejecutado el programa, creando un historial de la cantidad total ejecuciones y que salida de datos se produjo al usar las distintas funciones.

Para facilitar el ingreso de datos el programa crea un archivo binario en el cual se guardan puntos geográficos para su posterior uso. Estos puntos son almacenados y nombrados por el usuario en una estructura adecuada. El archivo sirve como base de datos para almacenar ubicaciones y evitar la pérdida de tiempo en el ingreso de datos a posteriori. Permitiendo al usuario ingresar el nombre asignado al punto en lugar de sus coordenadas geográficas.

4.2 Interfaz con el usuario

El programa se creó como una aplicación de consola, la interfaz del programa se presenta en modo terminal. Se tuvo en cuenta que a pesar de la sencillez de este medio, la Calculadora Geoespacial sea amigable con el usuario sin entorpecer su funcionamiento. Esto se traduce en que la utilización del programa no resulta difícil al usuario no experto.

Al iniciar la Calculadora esta se presenta y arroja un menú de opciones en las cuales se expresan las operaciones que se pueden realizar. Se buscó limitar el número de preguntas que se realizan, y es por eso que para ingresar un dato angular, los grados, minutos, segundos y orientación se leerán de una única vez como una cadena de caracteres.

```

C:\QtSDK\QtCreator\bin\qtcreator_process_stub.exe
*****
CALCULADORA GEOESPACIAL
*****

MENU
*****
1.Conversion de coordenadas
2.Distancia Euclidiana entre dos puntos
3.Distancia entre dos puntos - Modelo Esferico - Formula de Haversine
4.Distancia entre dos puntos - Modelo Elipsoide - Formula de Vincenty
5.Comparacion de distancia: Haversine - Vincenty
6.Comparacion de distancia: Euclidiana - Haversine - Vincenty
7.Puntos almacenados
8.Salir

1

CONVERSION DE UNIDADES
//Ingrese los valores correspondientes grados, minutos, segundos y orientacion
Ingrese los datos correspondientes a la latitud
42 55 44.4071 S
Ingrese los datos correspondientes a la longitud
71 18 1.6822 O

RESULTADOS DE LA CONVERSION:
Datos Ingresados: 42.00* 55.00' 44.41'' S ; 71.00* 18.00' 1.68'' O

Modelo de Elipsoide          Modelo de Esfera          Diferencia
*****                      *****
X= 1497.246746                X= 1495.570655           -0.111945 %
Y=-4423.543757                Y=-4418.591827           -0.111945 %
Z=-4329.532504                Z=-4339.234458           0.224088 %
    
```

Figura 2. Captura de pantalla, entrada y salida de la conversión de coordenadas.

Nota del docente:

En primer año no se enseñan elementos de programación orientados a objetos ni a su uso en la creación de interfaces graficas del usuario (GUI).

5 Evaluación de Resultados

5.1 Pruebas Realizadas

En un primer caso de prueba se tomó un caso de referencia^[6] de transformación de coordenadas geográficas a coordenadas cartesianas en el espacio de la bibliografía. Y se compararon los valores obtenidos por el programa usando el modelo de Esfera y Elipsoide Terrestre con el punto de dato, los resultados se aprecian en la tabla 1.

Para la comparación de las funciones de cálculo de distancia se realizaron pruebas usando puntos arbitrarios y se compararon con algoritmos escritos en Java de los mismos métodos, en el sitio web de Movable Type^{[7][8]}. Se escogieron puntos arbitrarios y se aparearon de forma aleatoria para realizar las pruebas cuyos resultados se muestran en la tabla 2.

5.2 Resultados

Referencia: 42° 55' 44.4071" S, 71° 18' 1.6822" O

Tabla 1

Modelo	Coordenadas Calculadas		Coordenadas de Comparación ^[6]
	Esfera	Elipsoide	Geoide
X	1495.570655	1497.246746	1499.83388
Y	4418.591827	4423.543757	4431.18732
Z	4339.234458	4329.532504	4322.47746

Los datos expresados en la tabla 1 y tabla 2, están en kilómetros, que deriva de especificar los distintos radios de la Tierra en esa unidad.

Puntos arbitrarios utilizados:

1. 50° 3' 59" N, 5° 42' 53" O
2. 22° 7' 49" N, 39° 23' 16" E
3. 42° 55' 44" S, 71° 18' 1" O
4. 86° 10' 5" S, 168° 12' 13" E
5. 31° 16' 6" S, 90° 0' 17" O
6. 75° 20' 10" S, 75° 20' 10" E
7. 61° 16' 32" N, 140° 14' 14" O

Tabla 2

	Calculadora Geoespacial		Movable Type		Error	
	Haversine	Vincenty	Haversine ^[7]	Vincenty ^[8]	Haversine	Vincenty
1-3	12136,54482	12716,83656	12140	12110,25155	0,02846%	5,00886%
2-4	12731,6984	12650,31248	12730	12714,80964	0,01334%	0,50726%
7-4	16439,63209	16862,08637	16540	16517,38764	0,60682%	2,08688%
1-7	6990,080717	7077,94388	6990	7013,086507	0,00115%	0,92480%
1-6	14817,11663	15162,67459	15100	15071,752	1,87340%	0,60326%
4-6	1704,637267	1543,851208	1705	1711,940701	0,02127%	9,81865%
2-5	8020,53506	8094,627506	8021	8004,254096	0,00580%	1,12907%
5-7	11241,032367	11182,47132	11240	11213,51218	0,00981%	0,27682%

6 Conclusión y futuros pasos

Este trabajo representa la culminación con éxito del programa Calculadora Geoespacial, el cual dejó una gratificante experiencia al poder trabajar en la implementación de la programación y la matemática para resolver problemas reales. Siendo uno de los mayores retos crear un contexto de programación para evaluar las fórmulas matemáticas las cuales fueron el resultado de un trabajo de investigación. El uso y comparación de distintos modelos dentro y fuera del programa, deja apreciar la diferencia entre la realidad y las distintas representaciones.

Existen varias aplicaciones prácticas posibles para este programa. Entre estas podemos destacar la ejecución en un servidor dada la naturaleza del lenguaje, o bien, embebido en un sistema que recibe la localización de un módulo GPS. Y ser utilizado en aplicaciones para dispositivos móviles.

7 Referencias

[1] Wikipedia the free Encyclopedia – “*Haversine formula*”.

<http://en.wikipedia.org/wiki/Haversine_formula>

[2] Wikipedia the free Encyclopedia – “*Vincenty's formulae*”.

<http://en.wikipedia.org/wiki/Vincenty's_formulae>

[3] Thaddeus Vincenty, “*Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations*” - Survey Review vol XXIII N° 176, Abril 1975 pp (88 – 93).

[4] Amaya Jaramillo – Carrera Chicaiza, “*Sistema de monitoreo pasivo para vehículos mediante GPS*”.

Proyecto Final – Escuela Politécnica Nacional, Quito, Ecuador.

<<http://dspace.epn.edu.ec/bitstream/15000/9979/1/T11762%20CAP1.pdf>>

[5] Publicaciones Online Universidad de Barcelona. “*Astronomía esférica y mecánica celeste*”

<http://www.publicacions.ub.edu/liberweb/astronomia_esferica/material/version_pdf/Tomo%201/2.1%20Elipsoide%20terrestre.pdf>

[6] CPIAA Chubut - Sitio de los Agrimensores, Páginas Técnicas GPS. “*Conceptos geodésicos básicos*”

<http://www.agrimensoreschubut.org.ar/PTGPS/Conceptos%20Geodesicos%20.pdf>

[7] Movable Type Ltd – Information Design & Management – “*Calculate distance, bearing and more between Latitud/Longitude points*”.

<<http://www.movable-type.co.uk/scripts/latlong.html>>

[8] Movable Type Ltd – Information Design & Management - “*Vincenty’s Direct formula*”

<<http://www.movable-type.co.uk/scripts/latlong-vincenty-direct.html>>