ESSE 3610 – Geodetic Concepts

PROJECT 2

November 11, 2018

Yaseen Al-Taie 213996921
Hunter Schofield 214193627

Executive summary➔

This report provides an analysis and evaluation of the knowledge of various geodetic concepts, specifically focusing on the analysis of ellipsoidal geometry, the transformations between Cartesian and curvilinear coordinates, and geodetic positioning techniques. The completion of this lab will provide better appreciation and comprehension of relevant concepts for this course.

In the first section, the azimuth is calculated for provided points on the Earth's surface as modelled by the WGS1984 reference ellipsoid. Further, the length of the geodesic is calculated in this section, along with the maximum height along the path between the provided points.

The second section of the lab is based around the mathematics which defines spherical and ellipsoidal geometry. Specifically, we discuss and prove radius of curvature in the prime vertical and the first eccentricity in the second section.

In the third section we find the geocentric Cartesian coordinates when given the translation components from the geocentric of Clark 1866 reference ellipsoid and in the same section we Calculate its geodetic curvilinear coordinates in the WGS1984 reference ellipsoid for a satellite having the following geocentric coordinates at a time instant.

In the final section, we use geodetic coordinates (ϕ, λ) of a point, Q, using Puissant's short line equations, and given the geodetic coordinates of a terrain point P, which is referred to by the WGS1984reference ellipsoid. We also Calculate the reverse geodetic azimuth α_{QP}, and the ellipsoidal distance $S_{PQ.}$

General Labor ➔ ***Yaseen (50%) and Hunter (50%)***

Table of Contents

Question (1-a)

Introduction

Two major components of geodesy are: knowing where a point of interest is and knowing how to get there. While this may be a simple problem if worked out on a sphere, the reality is that the Earth is a much more complicated shape, which is better approximated as an ellipsoid. To tackle the problem defined in question 1, the WGS1984 reference ellipsoid is used in order to gain a more precise definition of the azimuths between the two given points, the shortest distance between the two points, and the highest elevation between the two points.

Methodology

The following approach to this problem employs Puissant's inverse solution. First, an initial value for the azimuth angle between point A and B is calculated.

$$\alpha_{ij_k} = \tan^{-1}\left[\frac{\dfrac{\Delta\lambda\ N_j}{\sec\phi_j}}{\dfrac{\Delta\phi\ M_i}{\left(1 - \dfrac{3e^2 \sin\phi_i \cos\phi_i\ \Delta\phi}{2(1 - e^2 \sin^2\phi_i)}\right)}}\right] \qquad (1)$$

Then, this initial value can be used to determine the length of the geodesic (distance between points along curve).

$$S_{ij_k} = \frac{\Delta\lambda\ N_j}{\sec\phi_j\ \sin\alpha_{ij_k}} \qquad (2)$$

With these initial values calculated, we have the basis for an iteration. Using these values in an iteration of approximately 4 steps gives a solution with precision greater than 1 part in 1 billion (i.e. solution is iterated until difference is less than 10^{-9}). The following variables, T1 and T2, are intermediary and are only used to find the next iteration of the azimuth and length of the geodesic.

$$T_1 = \frac{\Delta\lambda\ N_j}{\sec\phi_j} + \frac{(S_{ij_k})^3}{6N_j^2}\sin\alpha_{ij_k} - \frac{(S_{ij_k})}{6N_j^2}\sin^3\alpha_{ij_k}\ \sec^2\phi_j \qquad (3)$$

$$T_2 = \Delta\phi\left[\frac{M_i}{\left(1 - \dfrac{3e^2 \sin\phi_i \cos\phi_i}{2(1 - e^2 \sin^2\phi_i)} \cdot \Delta\phi\right)}\right] + \frac{(S_{ij_k})^2 \tan\phi_i \sin^2\alpha_{ij_k}}{2N_i} + \frac{(S_{ij_k})^3 \cos\alpha_{ij_k} \sin^2\alpha_{ij_k} (1 + 3\tan^2\phi_i)}{6\ N_i^2} \qquad (4)$$

These intermediary values are then used to calculate the next iteration of azimuth and geodesic length according to the following.

$$\alpha_{ij_{k+1}} = \tan^{-1}\left(\frac{T_1}{T_2}\right) \quad (5)$$

$$S_{ij_{k+1}} = \frac{T_1}{\sin \alpha_{ij_k}} \quad (6)$$

To get the reverse azimuth – the azimuth from point B to A, we can apply the prior steps in reverse, or we can take the following.

$$\alpha_{ji} = \alpha_{ij} + \Delta\alpha + 180^{\circ} \quad (7)$$

Where Δα is defined as

$$\Delta\alpha = \Delta\lambda \operatorname{Sin}(\phi_m) \operatorname{Sec}\left(\frac{\Delta\phi}{2}\right) + \frac{\Delta\lambda^3}{12}\left(\operatorname{Sin}(\phi_m) \operatorname{Sec}\left(\frac{\Delta\phi}{2}\right) - \operatorname{Sin}^3(\phi_m) \operatorname{Sec}^3\left(\frac{\Delta\phi}{2}\right)\right) \quad (8)$$

Finally, to compute the highest point between point A and B, we use the calculated length of the geodesic to solve for the difference in latitude and longitude from the mean latitude and longitude between point A and B. The equation to solve for latitude and longitude work out to the following.

$$\Delta\phi_k = \left[\frac{S_{ij} \cos \alpha_{ij}}{M_i}\right] \quad (9)$$

$$\Delta\lambda_k = \left[\frac{S_{ij} \sin\alpha_{ij}}{N_m \cos \phi_m}\right] \quad (10)$$

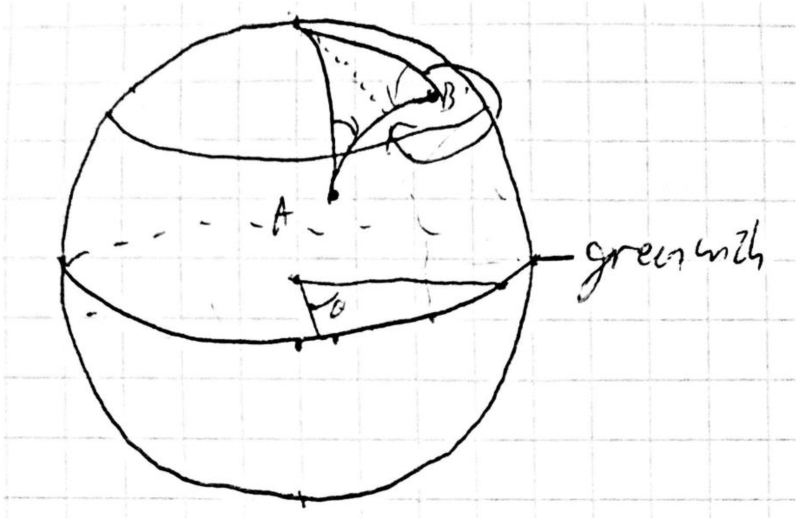
Processing Inputs

The static inputs for this problem are defined by WGS1984 reference ellipsoid, which has a semi-major axis of 6,378,137 meters, and a semi-minor axis of 6,356,752.3142 meters. The variable inputs for this problem are the latitude and longitude values for the points A and B. For group number 9, the values calculated for the latitude and longitude of point A in degrees is 43.7° and 280.367° respectively. The values calculated for the latitude and longitude of point B in degrees is 46.4° and 350.533° respectively.

Processing Outputs & Analysis

Applying the methodology yields an azimuth at point A to B of 60.84° and an azimuth at point B to A of 293.72°. The length of the geodesic from point A to B (i.e. the shortest surface length) determined is 5349198 meters. Finally, the highest point on the path between these points occurs at a latitude of 45.2° North and longitude of 328° East.

By inspection, these values seem correct. The following 3D illustration of the Earth with these points depicted reveals the illustrated azimuth angles to well represent the calculated values.



Further, we know that the length of the geodesic must be fairly close to the value of the arc length for the same points mapped onto a sphere. A quick calculation reveals that the arc length of these points on a sphere is equal to 5340135 meters which is fairly close to the value determined on the WGS1984 ellipsoid, which suggests the calculation should be valid.

Software Structure

The software structure is a straightforward script with two helper functions, one designed to make converting to radians easer, and one designed to make the iteration component easier. The remainder of the script is formula setup used to solve Puissant's inverse formula.

The full software used for this question can be viewed in appendix A.

Question (2-b)

Introduction ➔

Applying the methodology of trigonometry to show that the radius of curvature in the prime vertical. Trigonometry is a methodology for finding some unknown elements of a triangle (or other geometric shapes) provided the data includes an enough linear and angular measurements to define a shape uniquely. Also, applying the concept of polar flattening to show that for the first eccentricity (e^2) of the ellipsoid of rotation at any point of geodetic latitude.

Methodology, processing input, Analysis and discussion ➔

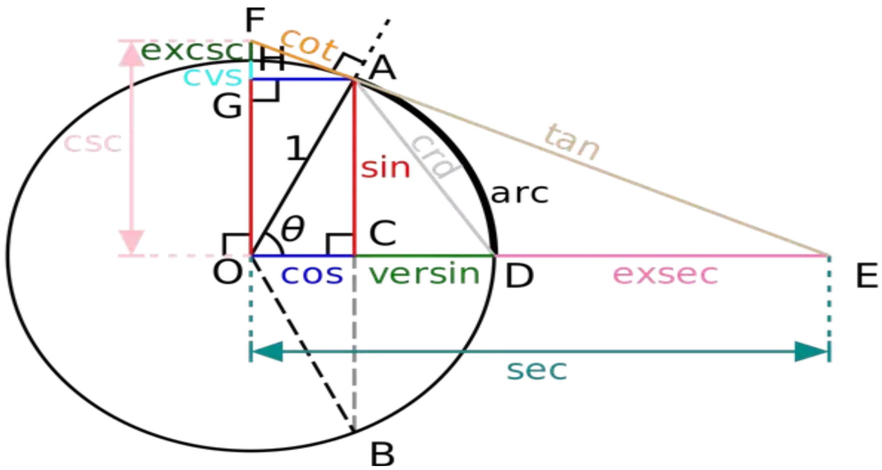
To prove the following

$$N = \frac{a}{\sqrt{1 - e^2 sin^2\phi}} \qquad (1)$$

We first consider the equation of an ellipse:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \qquad (2)$$

where from this relationship, by using geometric trigonometry we differentiate and get the equation of a tangent line to the curve at point A in figure 1



$$\frac{\partial y}{\partial x} = -\frac{b^2}{a^2}\frac{x}{y} \qquad (3)$$

$$-\cot\phi = \frac{\partial y}{\partial x} = \frac{\Delta y}{\Delta x} = \frac{rise}{run} \qquad (4)$$

From (3) and (4), we can find the equation of the normal line by taking the negative reciprocal

$$tan\phi = \frac{a^2}{b^2}\frac{y}{x} \qquad (5)$$

From the properties of an ellipse we know that

$$1 - e^2 = \frac{b^2}{a^2} \qquad (6)$$

Therefore, we get

$$b = a\sqrt{1 - e^2} \qquad (7)$$

And rearranging (5) we get that

$$y = \sqrt{1 - e^2} \tan\phi \, x \qquad (8)$$

Therefore, substituting (7) for b and (8) for y

$$\frac{x^2}{a^2} + \frac{(1 - e^2)^2 \tan^2\phi \, x^2}{a^2(1 - e^2)} = 1 \qquad (9)$$

simplifying we get

$$\frac{x^2}{a^2} + \frac{(1 - e^2) \tan^2\phi \, x^2}{a^2} = 1 \qquad (10)$$

factoring x, we get

$$x^2 \left(\frac{1}{a^2} + \frac{(1 - e^2) \tan^2\phi}{a^2} \right) = 1 \qquad (11)$$

which is also equal to,

$$x^2 \left(\frac{1 + (1 - e^2) \tan^2\phi}{a^2} \right) = 1 \qquad (12)$$

multiplying both sides by the reciprocal of the content in brackets

$$x^2 = \frac{a^2}{1 + (1 - e^2) \tan^2\phi} \qquad (13)$$

which is also equal to

$$x^2 = \frac{a^2}{1 + (1 - e^2)\left(\frac{\sin^2\phi}{\cos^2\phi} \right)} \qquad (14)$$

Rearranging (14), we get

$$x^2 = \frac{a^2 \cos^2\phi}{\cos^2\phi + (1 - e^2) \sin^2\phi} \qquad (15)$$

and by using trigonometric identities

$$x^2 = \frac{a^2 \cos^2 \phi}{(1 - \sin^2 \phi) + (1 - e^2) \sin^2 \phi} \qquad (16)$$

and simplifying (16), we get

$$x^2 = \frac{a^2 \cos^2 \phi}{1 - e^2 \sin^2 \phi} \qquad (17)$$

Squaring both sides of (17), we get

$$x = \frac{a \, cos \, \phi}{\sqrt{1 - e^2 \sin^2 \phi}} \qquad (18)$$

And from figure 1 and using geometry, we know that,

$$x = N \cos \phi \qquad (19)$$

Substituting (18) in to (19) for x, we get

$$\frac{a \, cos \, \phi}{\sqrt{1 - e^2 \sin^2 \phi}} = N \cos \phi \qquad (20)$$

Which finally when diving both sides by $\boldsymbol{cos\ \phi}$ gives us

$$\boldsymbol{N = \frac{a}{\sqrt{1 - e^2 sin^2 \phi}}} \qquad (\mathbf{20})$$

And to prove Eccentricity
We first consider the polar flattening which is defined by

$$f = \frac{a - b}{a} \qquad (21)$$

When (21) is rearranged we get,

$$\frac{a}{b} = 1 - f \qquad (22)$$

Eccentricity is also defined by

$$e^2 = \frac{a^2 - b^2}{a^2} \qquad (23)$$

When (23) is rearranged we get

$$e^2 = 1 - \frac{b^2}{a^2} \qquad (24)$$

Substituting (22) in (24) we get,

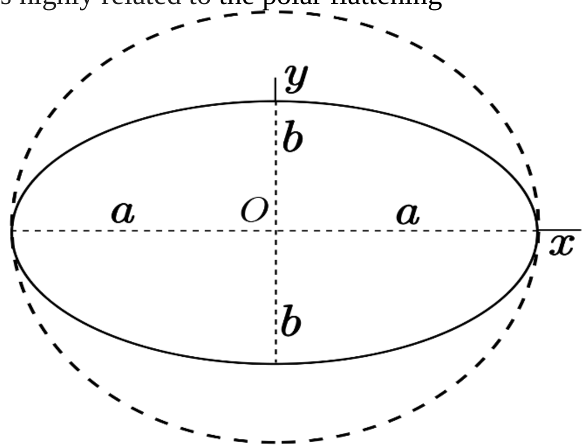
$$e^2 = 1 - (1 - f)^2 \qquad (25)$$

And finally rearranging (25) we get,

$$\boldsymbol{e^2 = 2f - f^2} \qquad (\mathbf{26})$$

Summary➔

After doing some proofs and differentiate, found out that geometric trigonometry is an accurate method to find any line or point in any shape. And in order to obtain the eccentricity, the polar flattening should take in consideration as it's a way to measure of the compression of a circle or sphere along a diameter to form an ellipse or an ellipsoid of revolution (spheroid) respectively. As seen in the figure, this shape provides us with the eccentricity as its highly related to the polar flattening



Question (3)

Introduction ➔

Since the Earth is flattened at the poles and bulges at the Equator, geodesy represents the figure of the Earth as an oblate spheroid. The oblate (spheroid and ellipsoid), is an ellipsoid of revolution obtained by rotating an ellipse about its shorter axis. Given the Clark 1866 geocentric reference ellipsoid we are supposed to calculate the geodetic curvilinear coordinates in the WGS84 reference ellipsoid for a satellite having the following geocentric coordinates at a time instant and calculate geocentric Cartesian coordinates.

Methodology, processing input and Analysis ➔ ***Part (a)***

The following geodetic coordinates for a terrain station, which have been converted to degrees and modified to reflect the values assigned to group 9, are given as follow.

$\phi = 44.295^\circ\, N$
$\lambda = 90.89^\circ\, E$
$h = 260.26\, m$

Also, it is known to us from Clark 1866 reference ellipsoid that the translation components are the following

$$X_0 = -25.82m$$
$$Y_0 = 168.10m$$
$$Z_0 = 167.31m$$

Using the semimajor and semi-minor values for the Clarke 1866 reference ellipsoid, it is possible to determine the following components.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (N+h)\cos\phi\, cos\,\lambda \\ (N+h)\cos\phi\, sin\,\lambda \\ \left(N\dfrac{b^2}{a^2}+h\right)\sin\phi \end{bmatrix}$$

Here, h is the height from ellipsoidal surface to a point on, above or below the Earth and normal to the eellipsoidal surface.

We know that ‘a’ the semi-major axis of the earth is 6.378137 x 10^6 m
And the ‘b’ the semi-minor axis is 6.3567523142 x 10^6 m

Now referring to (polar flattening) and substituting the values for a and b, we get

$$f = \frac{a-b}{a} = \frac{6.378137 \text{ x } 10^6 - 6.3567523142 \text{ x } 10^6}{6.378137 \text{ x } 10^6} = 3.3528106 \times 10^{-3}$$

And from there, we use (polar flattening) to find eccentricity

$$e^2 = 2f - f^2 = 2(3.3528106 \times 10^{-3}) \times (3.3528106 \times 10^{-3})^2 = 6.69437986 \times 10^{-3}$$

we now find N by using ➔

$$N = \frac{a}{\sqrt{1 - e^2 sin^2\phi}} = \frac{6.378137 \times 10^6}{\sqrt{1-(6.69437986 \times 10^{-3})^2 \times sin^2(43.545306944)}}$$

$$N = 6.3781625981 \times 10^6 m$$

Using N we can figure out X, Y, and Z by using the transformation coordinates ➔

$$x = (N + h)\cos\phi\cos\lambda$$

$$x = (6.3781625981 \times 10^6 + 260.26) \times \cos(44.295) \times \cos(90.89)$$

$$\mathrm{x} = -71082.6122\ \mathrm{m}$$

$$y = (N + h)\cos\phi\sin\lambda$$

$$y = (6.3781625981 \times 10^6 + 260.26) \times \cos(44.295) \times \sin(90.89)$$

$$y = 4572413.0192m$$

$$z = \left(N\frac{b^2}{a^2} + h\right)\sin\phi$$

$$z = \left(6.3781625981 \times 10^6 \times \frac{(6.3567523142 \times 10^6)^2}{(6.378137 \times 10^6)^2} + 260.26\right)\sin(44.295)$$

$$z = 4431591.2077m$$

Them an adjustment is done by adding the translation from the Clark 1866 reference ellipsoid,

$$x = x + X_o = -71056.7921m$$
$$y = y + Y_o = 4572581.1198m$$
$$z = z + Z_o = 4431758.5184m$$

Methodology, processing input and Analysis➔ Part (b)

Given

Satellite geocentric coordinates:

$$X_s = 4948685.566m$$
$$Y_s = -3249478.132m$$
$$Z_s = 3418646.589m$$

From the minor axis

$$p = \sqrt{{X_s}^2 + {Y_s}^2 + {Z_s}^2} = (N + h)cos\phi \qquad (\text{1986. Ek, Edward. p326.})$$

$$p = 6836354.3985m$$

Next, we find an initial value for ϕ using

$$\phi_0 = \tan^{-1}\left[\frac{z}{p}\left(\frac{1}{1-e^2}\right)\right]$$

Referring to WGS84 we get that e = 0.081819190842621, and then we can find an initial value for $\phi^{(0)}$

$$\phi_0 = (30.17148172°)$$

Next, we determine an initial value for N_0 by using the semimajor and semi-minor values for the WGS1984 reference ellipsoid.

$$N_0 = \frac{a^2}{\sqrt{a^2 cos^2\phi + b^2 sin^2\phi}}$$

$$N_0 = \frac{6378137^2}{\sqrt{6378137^2 cos^2(30.17148172°°) + 6356752.3142^2 sin^2(30.17148172°)}}$$

$$N_0 = 6397378.378\,m$$

Then, we determine an initial value for h_0 as

$$h_0 = \frac{p}{cos\phi} - N_0$$

$$h_0 = \frac{5920185.551}{\cos(30.17148172°)} - 6397378.378 = 450526.6478\,m$$

With these initial values, the basis for an iteration has been set up. It is possible to determine the next values for the iteration by defining the kth value as a function on the prior terms as follows.

$$N_k = N(\phi^{k-1})$$
$$h_k = h\left(\phi^{k-1}, N^{(k)}\right)$$
$$\phi_k = \phi\left(N^{(k)}, h^{(k)}\right)$$

After 4 iterations we find

$$\phi = 26.7173°$$
$$N = 6382456.6405m$$
$$h = 1271018.9821\,m$$

Now we find λ by using

$$\lambda = 2\tan^{-1}\left(\frac{Y_s}{X_s + P}\right)$$

$$\lambda = 2\tan^{-1}\left(\frac{-3249478.132}{4948685.566 + 6836354.3985}\right)$$

$$\lambda = -30.8302°$$

Therefore, we finally get the following

$$\begin{bmatrix}\phi\\ \lambda\\ h\end{bmatrix} = \begin{bmatrix}(26.7173°)\\ (-30.8302°)\\ (1271018.9821m)\end{bmatrix} = \begin{bmatrix}(26.7173°)\\ (329.1698°)\\ (1271018.9821m)\end{bmatrix}$$

Discussion and brief concepts on transformations ➔

Geocentric Cartesian coordinates are fixed to the rotating Earth, originating from the Earth's center. The *z*-axis points through the geographic North Pole (and coincides with the Earth's axis of rotation). Geodetic coordinates are often more convenient than spherical coordinates. In the geodetic coordinate system, the coordinates are altitude, longitude, and latitude. The geodetic latitude and longitude are the same latitude and longitude used in navigation and on maps. The geodetic and geocentric longitudes are the same. In geometry, curvilinear coordinates are a coordinate system for Euclidean space in which the coordinate lines may be curved. The conversion from curvilinear geodetic (λ, ϕ, h) to cartesian (x, y, z) coordinates needs some inputs ➔

The conversion from curvilinear geodetic (λ, ϕ, h) to Cartesian (x, y, z) coordinates is given by the well-known equations:

$$\begin{Bmatrix}x\\ y\\ z\end{Bmatrix} = \begin{Bmatrix}(N+h)\cos\phi\cos\lambda\\ (N+h)\cos\phi\sin\lambda\\ [N(1-e^2)+h]\sin\phi\end{Bmatrix} \quad (3)$$

$$X = (N + h_g)\cos(\phi_g)\cos(\lambda_g)$$
$$Y = (N + h_g)\cos(\phi_g)\sin(\lambda_g)$$
$$Z = [(1-e^2)N + h_g]\sin(\phi_g)$$

where $e^2 = 2f - f^2$ and N (North - South radius of curvature) is

$$N^2 = a^2/[1 - e^2\sin^2(\phi_g)]$$

Description of software structure ➔

The software built for this question is a straightforward script that sets up various equations in order to calculate the satellites geodetic curvilinear coordinates. There are two helper functions, one to facilitate conversions between degrees and radians, and one that eases the process of the iterations.

The full code for question 3 can be seen in Appendix B

Question (4)

Introduction➔

Calculate the geodetic coordinates (φ, λ) of point Q using Puissant's short line equations, the reverse geodetic azimuth and the ellipsoidal distance with given observations and approximate coordinates. There after we can use equations from the book (UNB) to get the reverse geodetic azimuth α_{QP}, and the ellipsoidal distance S_{PQ}.

Methodology, processing input, Analysis and Description of software structure ➔ (a, b and c)

Using Puissant's Equation we solve the direct problem as follows

We first find M using

$$M = \frac{a(1-e^2)}{(1-e^2\sin^2\phi)^{\frac{3}{2}}} \qquad \boldsymbol{N} = \frac{\boldsymbol{a}}{\sqrt{\boldsymbol{1 - e^2 sin^2\phi}}}$$

We then Approximate $\Delta\boldsymbol{\phi}$ using

$$\Delta\phi = \frac{s_{ij}}{N_i}\cos\alpha_{12} - \frac{s_{ij}^2}{2N_i^2}\tan\phi_i\, sin^2\alpha_{ij} - \frac{s_{ij}^3}{6N_i^3}\cos\alpha_{ij}\, sin^2\alpha_{ij}(1 + 3tan^2\phi_i) + \cdots$$

From there we can solve for $\Delta\boldsymbol{\phi}$ using

$$\Delta\phi = \left(\frac{s_{ij}\cos\alpha_{ij}}{M_i} - \frac{s_{ij}^2\tan\phi_i\, sin^2\alpha_{ij}}{2M_iN_i} - \frac{s_{ij}^3\cos\alpha_{ij}\, sin^2\alpha_{ij}(1+3tan^2\phi_i)}{6M_iN_i^2} * \ldots \left(1 - \frac{3e^2\sin\phi_i\cos\phi_i}{2\,(1-e^2\sin^2\phi_i)}\left(\frac{\Delta\phi}{1}\right)\right)\right)$$

And find $\boldsymbol{\phi_2}$ using

$$\phi_j = \phi_i + \Delta\phi$$

Now we can find N_2

From there we can find $\Delta\boldsymbol{\lambda}$ using

$$\Delta\lambda = \left(\frac{s_{ij}}{N_j}\sin\alpha_{ij}\, sec\phi_j\left(1 - \frac{s_{ij}^2}{6N_j^2}(1 - sin^2\alpha_{ij}\sec^2\phi_j\,)\right)\right)$$

And now we find $\boldsymbol{\lambda_2}$

$$\lambda_j = \lambda_i + \Delta\lambda$$

Thereafter, we find $\Delta\alpha$ using

$$\Delta\alpha = \left(\Delta\lambda \sin\phi_m \sec\frac{\Delta\phi}{2} + \frac{\Delta\lambda^3}{12}\left(\sin\phi_m \sec\frac{\Delta\phi}{2} - \sin^3\phi_m \sec^3\left(\frac{\Delta\phi}{2}\right) + \cdots\right)\right)$$

And to find α_{ji} we use

$$\alpha_{ji} = \alpha_{ij} + \Delta\alpha + 180°$$

Where

$$s_{ij} = 2Rsin^{-1}\left(\frac{l_{ij}}{2R_m}\right)$$

$$l_{ij} = \sqrt{\frac{\Delta r_{ij}^2 - \left(h_j - h_i\right)^2}{\left(1 + \frac{h_i}{R_m}\right)\left(1 + \frac{h_j}{R_m}\right)}}$$

$$R_m = \frac{1}{2}\left(R_i(\alpha) + R_j(\alpha)\right)$$

$$R_i = \frac{M_i N_i}{M_i \sin{\alpha_{ij}}^2 + N_i \cos\alpha_{ij}^2}$$

$$R_j = \frac{M_j N_j}{M_j \sin{\alpha_{ij}}^2 + N_j \cos\alpha_{ij}^2}$$

$$\phi_m = \frac{1}{2}(\phi_i + \phi_j)$$

Givens at point P➔

$$\phi_i = 45.6725°N$$
$$\lambda_i = 66.04366°W$$
$$h_i = 155.52m$$

Approximate coordinates of Q➔

$$\phi_j = 45°32N$$
$$\lambda_j = 66.29806°W$$
$$h_j = 155.52m$$

Other Parameters given➔

$$\alpha_{ij} = 297°18'59.13"$$
$$Z_{PQ} = 80°44'46.83"$$
$$\rho_{PQ} = 25{,}363.823\ m$$

We first find $\Delta\phi$ and $\Delta\lambda$ (we have the values and the equations)
As, $\Delta\phi$ is given in iterations, which is equal to

$$\Delta\phi = -0.065593350115222$$
$$\Delta\phi = -0.065567237188703$$
$$\Delta\phi = -0.065567217530277$$
$$\Delta\phi = -0.065567217515478$$

So $\Delta\lambda$ is

$$\Delta\lambda = -0.290396704721729$$

Using the above then we compute in ***MATLAB*** an iterative algorithm to find that the result is after 4 iterations

$$\boldsymbol{\phi = 45.606932782484520°N}$$
$$\boldsymbol{\lambda = 65.753263295278273°E}$$

Part B
Using the following equation, we equate the reverse geodetic Azimuth

$$\alpha^E{}_{ij} = tan\left[\frac{N_j\Delta\lambda}{M_i\Delta\phi}cos\phi_j\left(1-\frac{3e^2sin2\phi_i}{4(1-e^2sin^2\phi_i)}\right)\right]^{-1}$$

Using ($\Delta\alpha$) we find

$$\Delta\alpha = -0.208251887580058$$

Using ($\alpha^E{}_{ij}$) we compute α_{QP}, which gives

$$\boldsymbol{\alpha_{QP} = 177.1077°}$$

Part C
Puissant's solution to the inverse problem on an ellipsoid➔ we can get the values $\Delta\phi$, M_i , ϕ_i and α_{ij} from MATLAB code

$$S^E{}_{ij} = \frac{\Delta\phi}{cos\alpha_{ij}}\frac{M_i}{1-\frac{3e^2sin2\phi_i\Delta\phi}{4(1-e^2sin^2\phi_i)}}$$

Computing S_{PQ}, which gives

$$\boldsymbol{S_{PQ} = 23.1335077}\ \textbf{km}$$

software structure➔
The software has been explained literally by using the functions, equations and values step by step (applied all the equations on MATLAB). Also, we did iteration for (diphi which it should be larger than the epsilon to obtain the values for $\boldsymbol{\phi\ and\ \lambda}$**). Check out appendix C**

Summary➔
In this section we are more familiar with the properties of the ellipsoidal geometry and transformations between Cartesian and curvilinear coordinates, and reduction of observations to the ellipsoid and the techniques involved in computing geodetic positions on the ellipsoid both direct and inverse problems. Also, we become capable of performing the relevant calculation and applications in practice. We also understand the geometric relationship of the ellipse with respect to the Cartesian.

Appendix A

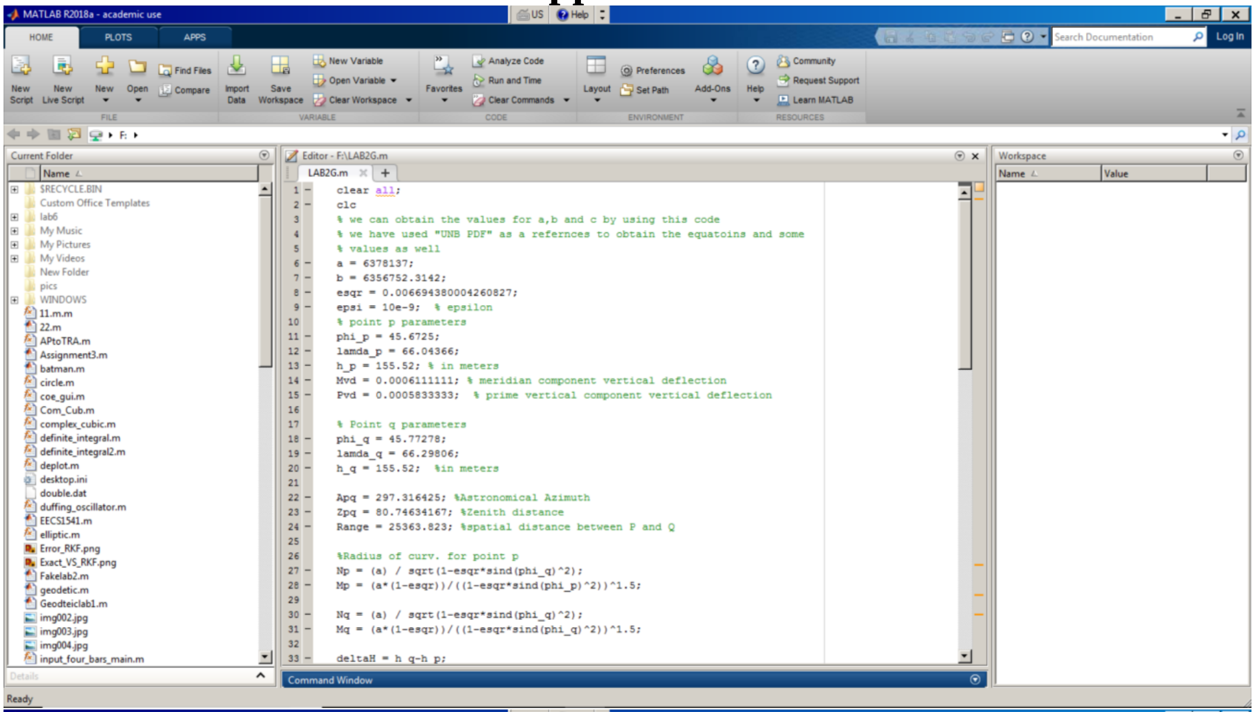
```python
import numpy as np

def toRad(theta):
  return theta * (np.pi / 180)

def solve():
  #based on WGS1984 reference ellipsoid
  a = 6378137 #meters
  b = 6356752.3142 #meters

  phi_A = toRad(43.7) #point A latitude
  phi_B = toRad(46.4) #point B latitude
  phi_M = (phi_A + phi_B) / 2 #mean of latitudes

  lam_A = toRad(280.367) #point A longitude
  lam_B = toRad(350.533) #point B longitude
  lam_M = (lam_A + lam_B) / 2 #mean of longitudes

  dl = lam_B - lam_A #delta longitude
  dp = phi_B - phi_A #delta latitude

  e = np.sqrt((a**2 - b**2)/a**2) #eccentricity

  Na = a / np.sqrt(1 - ((e**2)*(np.sin(phi_A)**2)))
  Nb = a / np.sqrt(1 - ((e**2)*(np.sin(phi_B)**2)))
  Nm = (Na + Nb) / 2

  Ma = a * (1 - e**2) / ((1 - ((e**2)*(np.sin(phi_A)**2)))**(3/2))
  Mb = a * (1 - e**2) / ((1 - ((e**2)*(np.sin(phi_B)**2)))**(3/2))
  Mm = (Ma + Mb) / 2

  #A function to get the next iteration
  def getNextIter(aijk, sijk):
    T1 = (dl*Nb / np.arccos(phi_B))
    T1 += (((sijk**3)/((Nb**2)))*(np.sin(aijk)))
    T1 -= ((sijk/(6 * (Nb**2)))*(np.sin(aijk)**3)*(np.arccos(phi_B)**2))

    T2 = dp*(Ma / (1 - ((3*(e**2)*np.sin(phi_A)*np.cos(phi_A)*dp) / (2*(1 - ((e**2)*(np.sin(phi_A)**2)))))))
    T2 += (((sijk**2)*np.tan(phi_A)*(np.sin(aijk)**2)) / (2*Na))
    T2 += (((sijk**3)*np.cos(aijk)*(np.sin(aijk)**2)*(1 + (3*(np.tan(phi_A)**2)))) / (6*(Na**2)))

    aijk1 = np.arctan(T1/T2)
    sijk1 = T1 / np.sin(aijk1)

    return (aijk1, sijk1)

  #determine initial values for a and s
  aij = np.arctan((dl*Nb / np.arccos(phi_B)) / ((dp*Ma) / (1 - ((3*(e**2)*np.sin(phi_A)*np.cos(phi_A)*dp) / (2*(1 - ((e**2)*(np.sin(phi_A)**2))))))))
  sij = (dl * Nb) / (np.arccos(phi_B) * np.sin(aij))

  #save the values to determine the change
  aijkm = aij
  sijkm = sij

  #get the new values
  aij, sij = getNextIter(aij, sij)

  #display the initial values
  print(aij)
  print(sij)

  #display the change in values from the previous iteration
  print(np.abs(aijkm - aij))
  print(np.abs(sijkm - sij))

  #while the change in iterations isnt below a certain threshold
  while np.abs(aijkm - aij) > 10**(-9):
    aijkm = aij
    sijkm = sij

    #get next iteration
    aij, sij = getNextIter(aij, sij)

    #display the values of the current iterations
    print("AIJ")
    print(aij)
    print(np.abs(aijkm - aij))

    print("SIJ")
    print(sij)
    print(np.abs(sijkm - sij))

  #determine the azimuth in the other direction
  da = (dl*np.sin(phi_M)*np.arccos(dp/2)) + (((dl**3)/12) * ((np.sin(phi_M)*np.arccos(dp/2)) - ((np.sin(phi_M)**3)*(np.arccos(dp/2)**3))))
  aji = aij + da + np.pi

  print("aji")
  print(aji)

  #highest point
  am = (aij + aji) / 2 #mean of azimuths

  #the result will be the difference in angle
  dp_h = (sij * np.cos(am)) / Mm
  dl_h = (sij * np.arccos(phi_M) * np.sin(am)) / Nm

  print("The highest point occurs at: \n")
  print("latitude: " + str(dp_h) + "\n")
  print("longitude: " + str(dl_h) + "\n")

if __name__ == "__main__":
  solve()
```

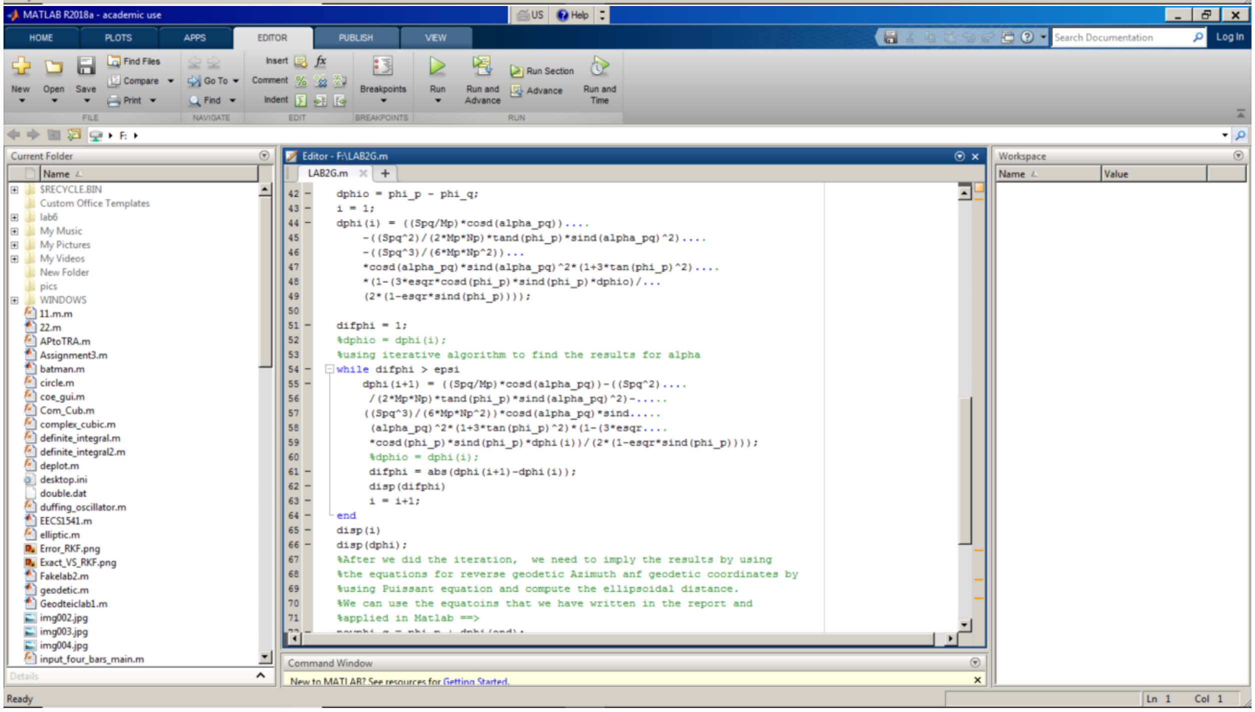
Appendix B

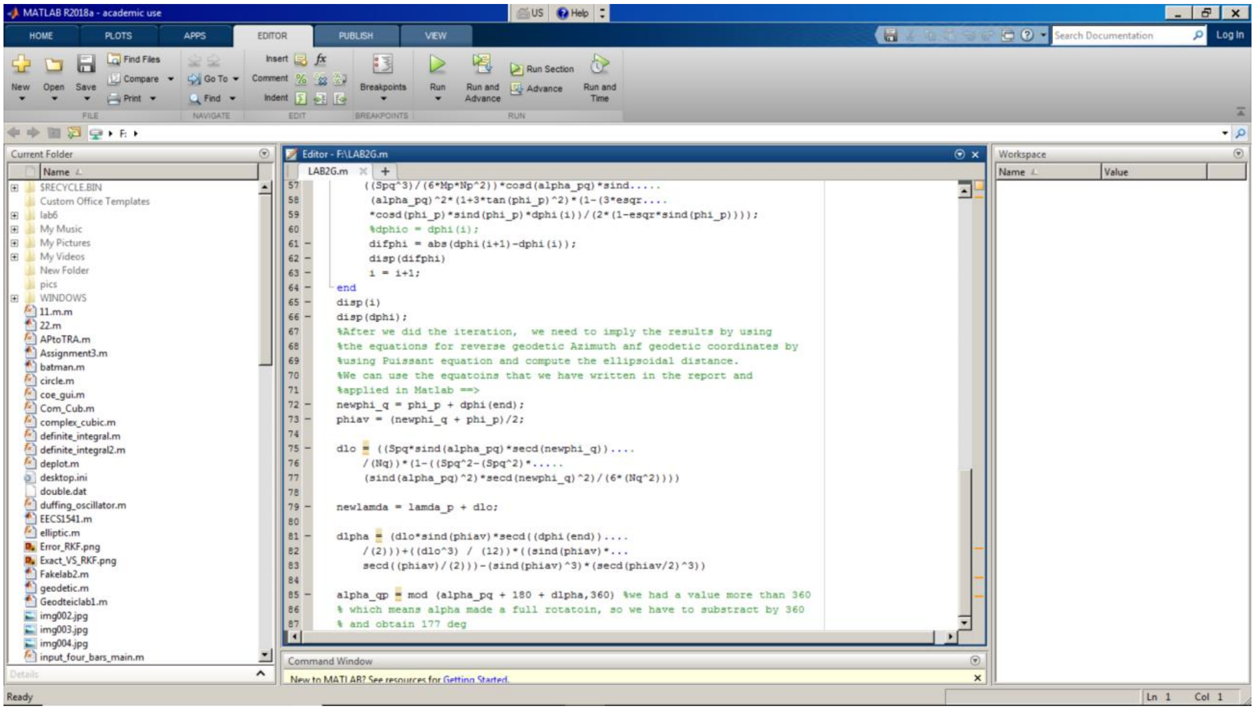
```python
import numpy as np

def toRad(theta):
    return theta * (np.pi / 180)

def solve():
    #based on clarke 1866 reference ellipsoid
    a = 6378206.4 #meters
    b = 6356583.8 #meters

    #calculate flatening and eccentricity
    f = (a - b) / a
    e = np.sqrt((a**2 - b**2)/a**2)

    x0 = -25.82
    y0 = 168.10
    z0 = 167.31

    phi = toRad(44.295)
    lam = toRad(90.89)
    h = 260.26

    N = a / np.sqrt(1 - ((e**2)*(np.sin(phi)**2)))

    #Calculate cartesian components and apply transformations
    X = (N + h)*np.cos(phi)*np.cos(lam) + x0
    Y = (N + h)*np.cos(phi)*np.sin(lam) + y0
    Z = (N*((b**2)/(a**2)) + h)*np.sin(phi) + z0

    #PART B
    #using WGS1984 reference ellipsoid, redifine parameters
    a = 6378137 #meters
    b = 6356752.3142 #meters

    f = (a - b) / a
    e = np.sqrt((a**2 - b**2)/a**2)

    #Satellite geocentric coordinates
    xs = 4948685.566
    ys = -3249478.132
    zs = 3418646.589

    #Find distance of satellite with respect to geocenter
    P = np.sqrt((xs**2) + (ys**2) + (zs**2))
    print(P)

    #setup iteration variables
    h = 0
    phi_0 = np.arctan((zs/P) * (1 / (1 - e**2)))
    N_0 = (a**2) / np.sqrt((a**2)*(np.cos(phi_0)**2) + (b**2)*(np.sin(phi_0)**2))
    h_0 = (P / np.cos(phi_0)) - N_0
```

```python

    #get the next iteration
    def getNextIter(p, n, h):
        _p = np.arctan(zs / P) / (1 - ((e**2)*n) / (n + h))
        _n = (a**2) / np.sqrt((a**2)*(np.cos(p)**2) + (b**2)*(np.sin(p)**2))
        _h = (P / np.cos(p)) - n

        return (_p, _n, _h)

    (phi_1, N_1, h_1) = getNextIter(phi_0, N_0, h_0)
    dp = np.abs(phi_1 - phi_0)
    dN = np.abs(N_1 - N_0)
    dh = np.abs(h_1 - h_0)

    #loop while there is a large enough difference between iterations
    while dp > 10**(-9) or dN > 10**(-9) or dh > 10**(-9):
        #set the last iteration values
        phi_0 = phi_1
        N_0 = N_1
        h_0 = h_1

        #get the next iteration values
        (phi_1, N_1, h_1) = getNextIter(phi_0, N_0, h_0)

        #find the difference to determine whether or not to keep iterating
        dp = np.abs(phi_1 - phi_0)
        dN = np.abs(N_1 - N_0)
        dh = np.abs(h_1 - h_0)

        #display the iteration results
        print(phi_1)
        print(N_1)
        print(h_1)
        print("\n")

    #convert back to degrees
    phi = (phi_1 * (180 / np.pi)) % 90
    h = h_1

    #determine the latitude and convert to degrees
    lam = (2*np.arctan(ys / (xs + P)) * (180 / np.pi)) % 360

    print("the geodetic curvilinear coordinates are:")
    print("latitdue: " + str(phi) + " degrees north")
    print("longitude: " + str(lam) + " degrees east")
    print("height: " + str(h) + " meters")

if __name__ == "__main__":
    solve()
```

Appendix C


Editor - F:\LAB2G.m
clear all;
clc
% we can obtain the values for a,b and c by using this code
% we have used "UNB PDF" as a refernces to obtain the equatoins and some
% values as well
a = 6378137;
b = 6356752.3142;
esqr = 0.00669438000426082 7;
epsi = 10e-9; % epsilon
% point p parameters
phi_p = 45.6725;
lamda_p = 66.04366;
h_p = 155.52; % in meters
Mvd = 0.0006111111; % meridian component vertical deflection
Pvd = 0.0005833333; % prime vertical component vertical deflection
% Point q parameters
phi_q = 45.77278;
lamda_q = 66.29806;
h_q = 155.52; %in meters
Apq = 297.316425; %Astronomical Azimuth
Zpq = 80.74634167; %Zenith distance
Range = 25363.823; %spatial distance between P and Q
%Radius of curv. for point p
Np = (a) / sqrt(1-esqr*sind(phi_q)^2);
Mp = (a*(1-esqr))/((1-esqr*sind(phi_p)^2))^1.5;
Nq = (a) / sqrt(1-esqr*sind(phi_q)^2);
Mq = (a*(1-esqr))/((1-esqr*sind(phi_q)^2))^1.5;
deltaH = h_q-h_p;
Command Window


Editor - F:\LAB2G.m
dphio = phi_p - phi_q;
i = 1;
dphi(i) = ((Spq/Mp)*cosd(alpha_pq))....
-((Spq^2)/(2*Mp*Np)*tand(phi_p)*sind(alpha_pq)^2)....
-((Spq^3)/(6*Mp*Np^2))...
*cosd(alpha_pq)*sind(alpha_pq)^2*(1+3*tan(phi_p)^2)....
*(1-(3*esqr*cosd(phi_p)*sind(phi_p)*dphio)/...
(2*(1-esqr*sind(phi_p))));
difphi = 1;
%dphio = dphi(i);
%using iterative algorithm to find the results for alpha
while difphi > epsi
dphi(i+1) = ((Spq/Mp)*cosd(alpha_pq))-((Spq^2)....
/(2*Mp*Np)*tand(phi_p)*sind(alpha_pq)^2)-.....
((Spq^3)/(6*Mp*Np^2))*cosd(alpha_pq)*sind.....
(alpha_pq)^2*(1+3*tan(phi_p)^2)*(1-(3*esqr....
*cosd(phi_p)*sind(phi_p)*dphi(i))/(2*(1-esqr*sind(phi_p))));
%dphio = dphi(i);
difphi = abs(dphi(i+1)-dphi(i));
disp(difphi)
i = i+1;
end
disp(i)
disp(dphi);
%After we did the iteration, we need to imply the results by using
%the equations for reverse geodetic Azimuth anf geodetic coordinates by
%using Puissant equation and compute the ellipsoidal distance.
%We can use the equatoins that we have written in the report and
%applied in Matlab ==>
Command Window
New to MATLAB? See resources for Getting Started.



References

1) Http://ljournal.ru/wp-content/uploads/2017/03/a-2017-023.pdf. (2017). doi:10.18411/a-2017-023