بسم الله الرحمن الرحيم

Palestine Polytechnic University

College of Engineering
Civil and Architecture Engineering Department

Graduation Project:

Geo Transform Version 2.0

Done by:

Abdullah R. Radwan

Supervisor:

Eng. Musab Shahin

Palestine Polytechnic University

Hebron – Palestine

2016

بسم الله الرحمن الرحيم

Palestine Polytechnic University

College of Engineering
Civil and Architecture Engineering Department

Graduation Project:

Geo Transform Version 2.0

Done by:

Abdullah R. Radwan

In accordance with the recommendation of project supervisor and acceptance of all examining committee members, this project has been submitted to the Department of Civil and Architectural Engineering in the college of Engineering and Technology in partial fulfillment of requirements of the department for degree of Bachelor of Surveying and Geomatics Engineering**.**

Signature of Project Supervisor **Signature of Department Chairman**

Name…………………………… **Name…………………………**

Hebron – Palestine

2016

بسم الله الرحمن الرحيم

Palestine Polytechnic University

College of Engineering
Civil and Architecture Engineering Department

Graduation Project:

Geo Transform Version 2.0

Done by:

Abdullah R. Radwan

Supervisor

Eng. Musab Shahin

A project is submitted in partial fulfillment of requirements for the degree of Bachelor of Engineering in Survey and Geomatics Engineering

Palestine Polytechnic University

Hebron – Palestine

2016

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1. Background

This project is Geodetic Transformation tool programmed using C# programming language. This tool has the following functionalities; the first one to transform from Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z), where the same ellipsoid is used. The second functionality is to transform Geographic Coordinates (λ,φ) to 2D projected (Grid) coordinates (Easting, Northing), the map projections that have been used in this transformation tool are: Cassini, Transverse Mercator, Universal Mercator (UTM). The third is to apply the reverse transformation of the first and second functionalities.

In addition, a separate datum transformation tool is built to calculate the datum transformation parameters using different datum transformation methods, as follow: similarity 7-parameters, Helmert 2D Conformal, Helmert 2D Affine, Helmert 3D Linearized and Molodensky transformation.
In the datum transformation tool three basic map projections are used; Cassini, Transverse Mercator(TM) and Universal Transverse Mercator (UTM). In Palestine there are four map projections and coordinate systems are used, these are: Palestine_1923 Grid, Palestine_1923 Belt, Palestine_1923 CS Israel Grid and Israel_TM_Grid, where each coordinate system has its own projection parameters.

Also, a new tool had been added to Geo-Transform V 2.0 that the previous version didn't have, a conversion tool. The new tool basically convert from decimal to radian or from decimal to degrees or from degrees to radian and the vice versa.

1.2. Objectives

This project deals with C# programming language. This tool should be able to apply the following coordinate transformations.

1. Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z).
2. Geocentric coordinates (X,Y,Z) to Geographic coordinates (λ,φ,h)
3. Geographic coordinates (λ,φ) to Projected coordinates (Easting, Northing), for Cassini, Transverse Mercator (TM) and Universal Transverse Mercator (UTM).
4. Projected coordinates (Easting, Northing) to Geographic coordinates (□,φ) using Cassini, Transverse Mercator (TM) and UTM.
5. Calculate the datum transformation parameters using different transformation method; similarity 7-parameters, Helmert 2D Conformal, Helmert 2D Affine, Helmert 3D Linearized and Molodensky transformation.

1.3. Previous Studies

This software Geo-Transform V 2.0 is developed and updated from Geo-Transform software which was programmed using Microsoft Visual Basic 6.0, and it was designed and developed by Eng. Sumia Zahda and Eng. Manar Al-Jabari as their graduation project in 2008.

There are many software available to apply these calculations; such as ArcView, ArcGIS, Autodesk Land Desktop, ERMapper, ERDAS, . . . etc. but these software are expensive for a single user. In addition, these software have large capabilities and steps of work because the major task is not the coordinates transformations, but coordinates transformations is a secondary task, used to enhance the performance of the software.

1.4. Project Plan

The project works will be achieved by the following steps:

1. Literature review
2. Research Plan, determining the problem solving scope.
3. Transform from Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z) and in the opposite direction. `
4. Transform from Geographic coordinates (λ,φ) to Projected coordinates (Easting, Northing) and in the opposite direction, using three basic map projection; Cassini, Transverse Mercator (TM) and Universal Transverse Mercator (UTM).
5. Four basic coordinates systems are to be used: Palestine_1923 Grid, Palestine_1923 Belt, Palestine_1923 CS Israel Grid and Israel_TM_Grid. And insert false Easting, false Northing, Latitude of origin, Central Meridian and ellipsoid for each of coordinate system, are to be define in the software.
6. Calculate the datum transformation parameters using different transformation method; similarity 7-parameters, Helmert 2D Conformal, Helmert 2D Affine, Helmert 3D Linearized and Molodensky transformation.
7. Convert from decimal to radian to degrees and vice versa.
8. Programming datum transformation tool as a separate software.

Table (1.1): Time Table

Stage	Required Time /Week	Weeks													
		First Semester							Second Semester						
		2	4	6	8	10	12	14	2	4	6	8	10	12	14
Planning & Collecting Data	6														
System Analysis	4														
Problem Definition	4														
System Design	2														
Building the System and Programming	12														
Examine the system	4														
Documentation	28														

1.5. Methodology

The project has the following scope:

Chapter1: this chapter introduces the project.

Chapter2: this chapter explains the coordinates systems in this project.

Chapter3: this chapter describes the datum transformations, methods and applications.

Chapter4: this chapter shows the mathematical method to convert between the three different systems (Decimal degrees, Radian, Degrees/minutes/seconds).

Chapter5: this chapter shows how the user can use the software Geo Transform V 2.0.

CHAPTER 2

COORDINATE SYSTEMS

2.1. Introduction

2.2. Spherical Coordinate

2.3. Ellipsoidal Coordinate

2.4. Projected Coordinate

COORDINATE SYSTEMS

2.1. Introduction

Coordinate system is a system to determine location on the surface of the earth, different units and length to the angular distance have been used. The mathematical figure of the earth is applied to the classical definition of the geoid defined as equipotent surface of the earth gravitation field that nearly coincides with the mean sea level (MSL).

A reference surface is chosen so that reductions are applied to the surface. At the beginning this surface was defined as a sphere with a radius R (approximately R=3678km), later it was defined as a rotational ellipsoid, the circle on the equator with radius (a) and the distance from the center to the north or south pole is (b). (a) is called the semi major axis and (b) is called the semi minor axis, while always (a) is larger than (b).

Figure (2.1): The Earth [19]

2.2. Spherical Coordinate

Considering the earth as a sphere with radius R, the position of the point is defined by polar coordinate (λ,φ,h), or by Geocentric coordinates (X,Y,Z) with origin at the earth center.

Where;

λ = longitude; defined as the angular distance on the equator from Greenwich meridian to the local meridian.

φ= latitude; defined as the angular distance long the meridian from the equator to the point.

h = ellipsoid height; defined as the distance along the normal from the ellipsoid surface to the point.

The conversion between the different coordinates can be calculated as follows:

$$X = (R + h) \cos\varphi \sin\lambda \quad (2.1)$$

$$Y = (R + h) \cos\varphi \sin\lambda \quad (2.2)$$

$$Z = (R + h) \sin\varphi \quad (2.3)$$

The reverse formulas are:

$$r = \sqrt{X^2 + Y^2 + Z^2} \quad (2.4)$$

$$h = r - R \quad (2.5)$$

$$\varphi = \tan^{-1}\frac{Z}{\sqrt{(X^2+Y^2)}} \quad (2.6)$$

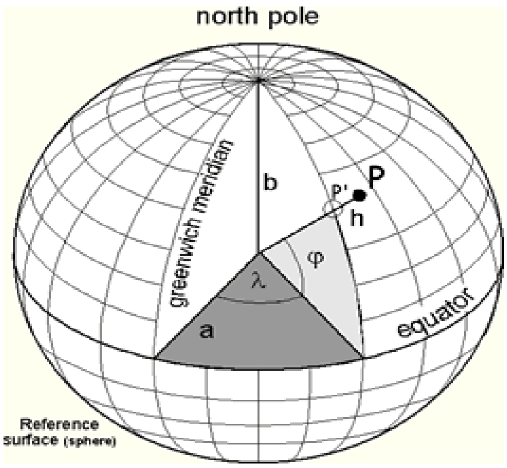

Figure (2.2): Sphere as the earth figure [8]

2.3. Ellipsoidal Coordinates

It is a three-dimensional orthogonal coordinate system (λ,φ,h) that generalizes the two-dimensional elliptic coordinate system. Unlike most three-dimensional orthogonal coordinate systems that feature quadratic coordinate surfaces, the ellipsoidal coordinate system is not produced by rotating or projecting any two-dimensional orthogonal coordinate system.

2.3.1. Ellipsoidal earth figure

In classic definitions; earth is considered to be an ellipsoid with semi major axis (a) & semi minor axis (b), the other basic parameters can be calculated using the basic axis, table (2.1) show selected reference ellipsoids. Table (2.2), show the basic defining parameters of an ellipsoid.

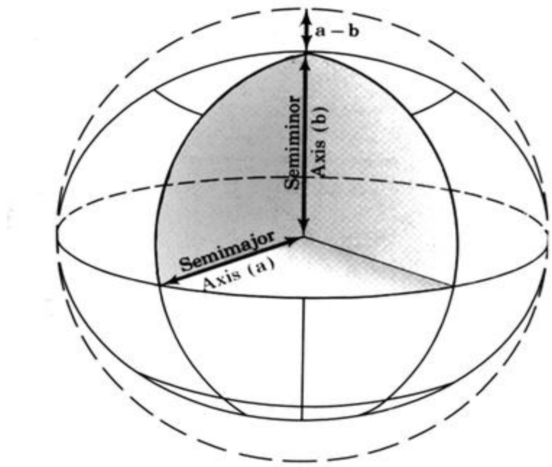

Figure (2.3): Ellipsoid as the figure of the earth [16]

Table (2.1): Selected reference Ellipsoid [15]

Ellipse	Semi-Major axis (meter)	1/Flattening
WGS 60	6378165.0	298.3
WGS 66	6378145.0	298.25
WGS72	6378135.0	298.26
WGS 84	6378137.0	298.257223563
South American 1969	6378160.0	298.25
Krassovky	6378245.0	298.3
International	6378388.0	297.0
Clarke 1866	6378206.4	294.978698
Clarke 1880	6378249.145	293.465
GRS 1975	6378140.0	298.257
GRS 1980	6378137.0	298.2572221

Table (2.2): The relation between the different parameters [15]

Notation	a/b	e^2	e'^2	f	n
a					$\frac{c(1-n)}{1+n}$
b		$a(1-e^2)^{1/2}$	$\frac{a}{(1+e^{`2})^{1/2}}$	$a(1-f)$	
$\frac{a}{b}$				$\frac{1}{1-f}$	
$\frac{b}{a}$		$(1-e^2)^{1/2}$	$\frac{1}{(1+e^{`2})^{1/2}}$	$(1-f)$	$\frac{1-n}{1+n}$
c	$\frac{a^2}{b}$	$\frac{a}{(1-e^2)^{1/2}}$	$a(1+e^{`2})^{1/2}$	$\frac{a}{1-f}$	
e^2	$\frac{a^2-b^2}{a^2}$		$\frac{e^{`2}}{1+e'^2}$	$f(2-f)$	$\frac{4n}{(1+n)^2}$
e'^2	$\frac{a^2-b^2}{b^2}$		$\frac{f(2-f)}{(1-f)^2}$		$\frac{4n}{(1-n)^2}$
f	$\frac{a-b}{a}$	$1-(1-e^2)^{1/2}$	$1-(1+e^{`2})^{-1/2}$		$\frac{2n}{1+n}$
n	$\frac{a-b}{a-b}$	$\frac{1-(1-e^2)^{1/2}}{1+(1-e^2)^{1/2}}$	$\frac{(1+e^{`2})^{1/2}-1}{(1+e^{`2})^{1/2}+1}$	$\frac{f}{2-f}$	

In modern definition the ellipsoid is defined as an equipotential surface. The normal potential (U) on the reference ellipsoid is equal to the geopotential (W) on the geoid. The total mass of the reference ellipsoid is equal to that of the earth, and reference ellipsoid is rotating around its minor axis at the same angular velocity as the earth rotation.

Many ellipsoids were defined by physical definition on the principle of normal gravity. Examples of physically defined ellipsoids are GRS 67, GRS 80 and WGS 84. The defining parameters for GRS 80 ellipsoid are shown in table (2.3).

Table (2.3): GRS 80 ellipsoid parameters [5]

Notation	Constant	Unit	Numerical value
a	Semi-major axis	m	6378137.000
GM	Product of G and total mass M	$m^3 s^{-2}$	0.3986005 E15
J2	Dynamic from factor $\frac{C-A}{Ma^2}$		0.00108263
ω	Angular velocity	s^{-1}	0.72921151 E-4
b	Semi-minor axis	m	6356752.3141
f	Geometrical flattening		1/298.257222101 = 0.003352810681
e^2	First eccentricity squared		0.006694380023
$e^{`2}$	Second eccentricity squared	s^{-1}	0.006739496775
U。	Normal potential on the ellipsoid	$m^2 s^2$	62636860.850
ɤp	Normal gravity on the poles	Gal	983.21863685
ɤe	Normal gravity on the equator	Gal	978.03267715
f*	Gravity flattening		1/188.592417552 = 0.005302440112
k	(b.ɤp –a.ɤe)/(a.ɤe)		0.001931851353
m	$\omega^2 a^2 b/(GM)$		0.003449786003
ɤ45	Normal gravity at altitude 45°	Gal	980.6199203
ɤ		Gal	979.7644656

The four parameters defining for WGS are shown in table (2.4).

Table (2.4) WGS ellipsoid parameters [21]

Parameter	Notation	Value
Semi-major axis	a	6378137.0 m
Reciprocal of Flattening	1/f	298.257223563
Angular Velocity of the Earth	ω	7292115.0*10 rad/s
Earth's Gravitational constant	GM	3986004.418*10 m^2/s^2

2.3.2. Geographic Coordinate

The most commonly used coordinate system today is the latitude, longitude & height. Equator & the prime meridian (Greenwich) are the reference planes used to define latitude and longitude.

The geodetic latitude (φ) of a point is the angle from the equatorial plane to the vertical direction of a line normal to the reference ellipsoid passing the point, the geodetic longitude (λ) of a point is the angle between a reference meridian (Greenwich) and the vertical plane passing through the point measure along the equator, both plans being perpendicular to the equatorial plane. The geodetic (ellipsoid or normal) height (h) at a point is the distance from the reference ellipsoid to the point in the direction normal to the ellipsoid. [5]

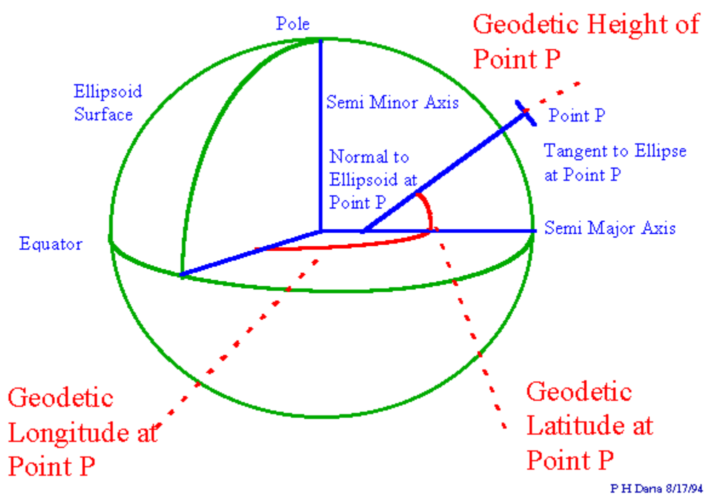

Figure (2.4): Geographic coordinate system [3]

2.3.3. Geocentric coordinates

It's a system of three dimensional earths centered reference system in which location are identified by their: X, Y and Z value.

The X axis is in the equatorial plane of intersects the prime meridian (Greenwich).

The Y axis is in the equatorial plane of intersects the +90^{o} meridian.

The Z axis coincides with the polar axis and is positive toward the North Pole.

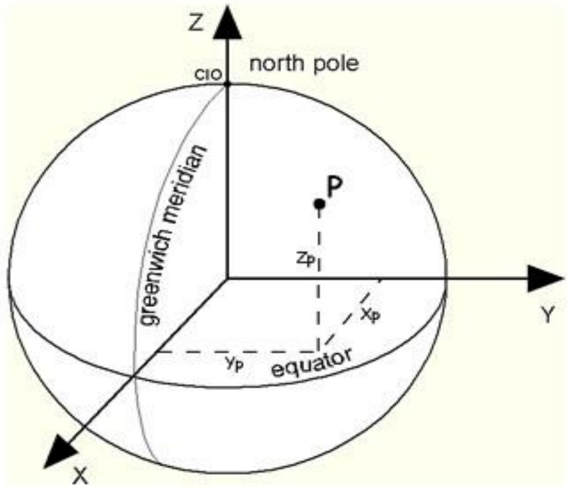

Figure (2.5): Geocentric system [8]

This system is not a grid system. The earth is modeled as a sphere or spheroid in a right handed X, Y, Z system

$$X = (N + h)\cos\varphi\cos\lambda \quad (2.7)$$
$$Y = (N + h).\cos\varphi.\sin\lambda \quad (2.8)$$
$$Z = ((1 - e).N + h).\sin\varphi \quad (2.9)$$

The inverse problem is solved in an iterative solution (Torge method): [9]

$$\lambda = \tan^{-1}\frac{Y}{X} \quad \text{(Does not need iteration)} \quad (2.10)$$

$$h = \frac{\sqrt{X^2+Y^2}}{\cos\varphi} - N \quad (2.11)$$

$$\varphi = \tan^{-1}\left(\frac{Z}{\sqrt{X^2+Y^2}}\left(1 - e^2\frac{N}{N+h}\right)^{-1}\right) \quad (2.12)$$

φ As initial value to start the iterative solution.

$$\varphi = \tan^{-1}\left(\frac{Z}{\sqrt{X^2+Y^2}}(1 - e^2)^{-1}\right) \quad (2.13)$$

2.3.4. Topocentric Coordinates

In the topocentric coordinates, we use the point of origin with known geographic coordinate P_o (λ,φ,h) or (X,Y,Z), the (X) direction in to the north, the (Y) direction is to the east and the (Z) direction is perpendicular on the (X, Y) plane.

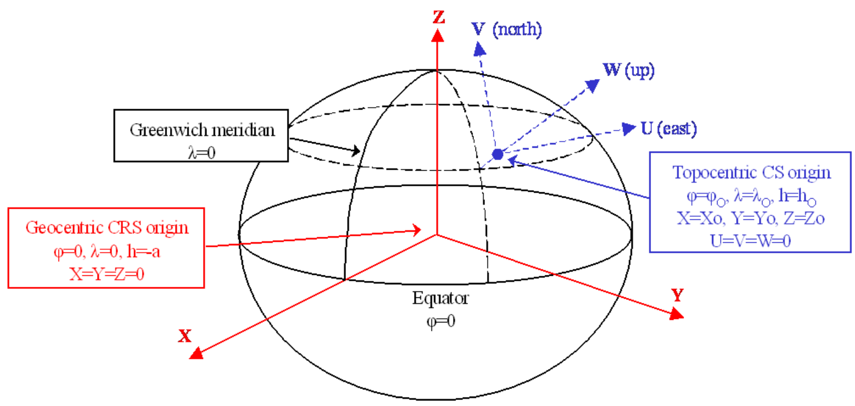

Figure (2.6): Topocentric coordinate system [13]

The position of the point is defined by the zenith (z_e), distance (s) and Azimuth (Az) measured clockwise from the north,

Where;

$$X = S \cos Az \sin z_e \quad (2.14)$$

$$Y = S \sin Az \sin z_e \quad (2.15)$$

$$Z = S \cos z_e \quad (2.16)$$

If geocentric coordinates are used

$$\mathrm{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \mathrm{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.17)$$

To convert from topocentric to geocentric coordinate, the following can be applied in matrix form.

$$\Delta X = \mathrm{Ax} \quad (2.18)$$

$$\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix} = \begin{bmatrix} -\sin\varphi_o \cos\lambda_o & -\sin\lambda_o & \cos\varphi_o \cos\lambda_o \\ -\sin\varphi_o \sin\lambda_o & \cos\lambda_o & \cos\varphi_o \sin\lambda_o \\ \cos\varphi_o & 0 & \sin\varphi_o \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.19)$$

$$X = X_{p0} + \Delta X \quad (2.20)$$

$$X = A^{-1}\Delta X = A^T \Delta X \quad (2.21)$$

2.4. Map Projection (Grid Coordinates System)

The basic idea at map projection is to convert from geographic (∅, λ) to local (E, N) Grid system and in the opposite direction, map projections are attempts to portray the surface of the earth or a portion of the earth on a flat surface. Some distortions of conformality, distance, direction, scale and area always result from this process. Some projections minimize distortions in some of these properties at the expense of maximizing errors in others. Some projections are attempts to only moderately distort all of these properties, the properties are: [14]

1. Conformalety: when the scale of a map at any point on the map is the same in any direction, the projection is conformal. Meridians (Lines of longitude) and parallels (lines of latitude) intersect at right angles. Shape is preserved locally on conformal maps.
2. Distance: A map is equidistant when it portrays distances from the center of the projection to any other place on the map.
3. Direction: A map preserves direction when azimuths (angles from a point on a line to another point) are portrayed correctly in all directions.
4. Scale: is the relationship between a distance portrayed on a map and the same distance on the earth.
5. Area: When a map portrays areas over the entire map so that all mapped areas have the same proportional relationship to the areas on the earth that they represent, the map is an equal-area map.

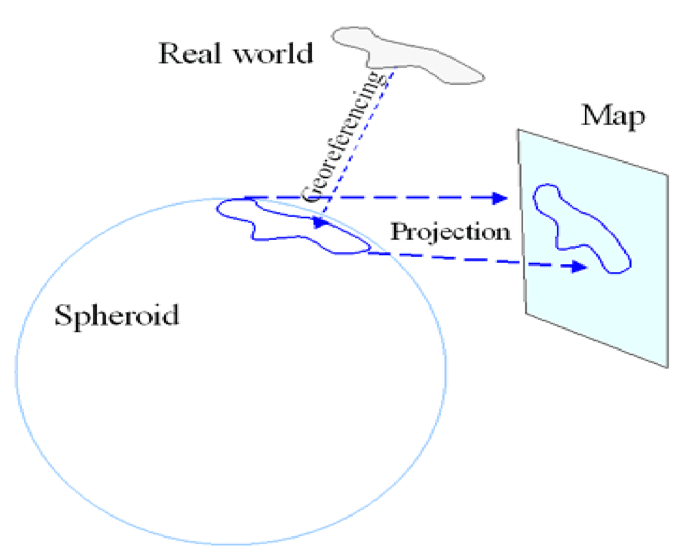

Figure (2.7): Map projection (from 3D to grid system) [18]

2.4.1. Map Projections General Classes

Map projections fall into four general classes depend on the methods of projection, and they are:

1. Cylindrical projections: result from projecting a spherical surface onto a cylinder.

a. When the cylinder is **tangent** to the sphere the contact is along a great circle (the circle formed on the surface of the earth by a plane passing through the center of the earth). [14]

Central meridian selected by mapmaker. Change in spacing of parallels less than that on Mercator projection. The equator always touches cylinder.

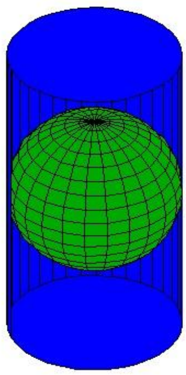

Figure (2.8): Projection of a sphere onto a cylinder (Tangent case). [17]

In the **secant** case, the cylinder touches the sphere along two lines

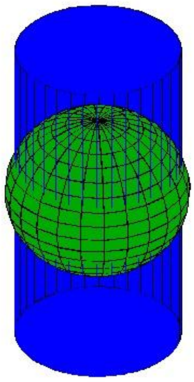

Figure (2.9): Projection of a sphere onto a cylinder (Secant case). [17]

b. When the cylinder upon which the sphere is projected is at right angles to the poles, the cylinder and resulting projection are **transverse**. [14]

Central meridian selected by mapmaker touches cylinder if the cylinder is tangent. This projection cam show whole earth, but the directions, distances and areas are reasonable accurate only within 15 degrees of the central meridian. No straight rhumb lines.

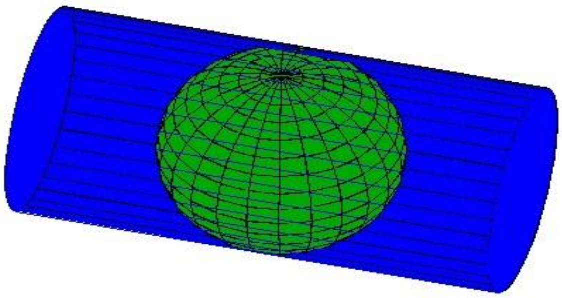

Figure (2.10): Transverse Projection of a sphere onto a cylinder (Tangent case). [17]

c. When the cylinder is at some other, non-orthogonal angle with respect to the poles, the cylinder and resulting projection is **oblique**. [14]

The great circle that touches cylinder is tangent. In this projection, shortest distances between points along line of tangency are straight lines.

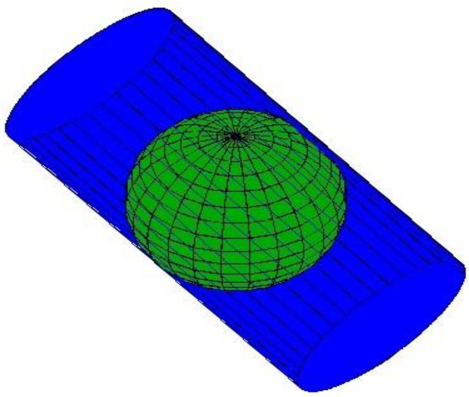

Figure (2.11): Oblique Projection of a sphere onto a cylinder (Tangent case). [17]

2. Conic Projections: result from projecting a spherical surface onto a cone.

In this projection, two standard parallels selected by mapmaker. Large scale map sheets can be joined at edges if they have the same standard parallels and scales.

a. When the cone is **tangent** to the sphere the contact is along small circle.

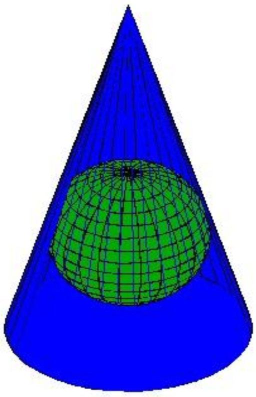

Figure (2.12): Projection of a sphere onto a cone (Tangent case). [17]

b. In the **secant** case, the cone touches the sphere along two lines; the lower line makes a greater circle than the upper line. [14]

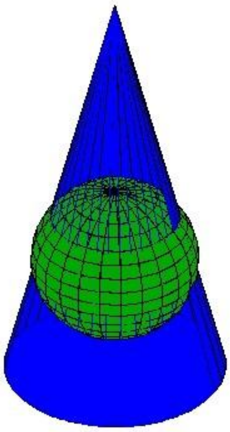

Figure (2.13): Projection of a sphere onto a cone (Secant case). [17]

3. Azimuthal Projections: result from projecting a spherical surface onto a plane.
   a. When a plane is **tangent** to the sphere the contact is at a single point on the surface of the earth. [14]

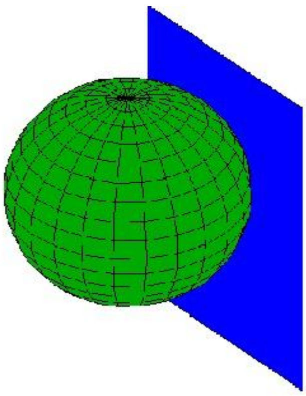

Figure (2.14): Projection of a sphere onto a plane (Tangent case). [17]

b. In the **secant** case, the plane touches the sphere along a small circle if the plane does not pass through the center of the earth, when it will touch along a great circle. [14]

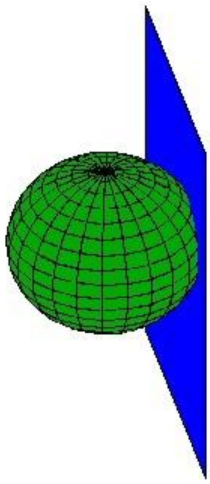

Figure (2.15): Projection of a sphere onto a plane (Secant case). [17]

4. Miscellaneous Projections: include unprojected ones such as rectangular latitude and longitude grids and other examples of that do not fall into the cylindrical, conic or azimuthal categories. [14]

2.4.2. Transverse Mercator

It is known as Gauss-Kruger and similar to the Mercator, except that the cylinder is longitudinal along a meridian instead of the equator. The result is a conformal projection that does not maintain true directions exclude small areas. The central meridian is placed in the center of the region of interest. This centering minimizes distortion of all properties in that region. This projection is best suited for north-south shaped areas. Distances are true only along the central meridian selected by the mapmaker or else along two lines parallel to it, but all distances, directions, shapes, and areas are reasonably accurate within 15° of the central meridian. Distortion of distances, directions, and size of areas increases rapidly outside the 15° band. Because the map is conformal, however, shapes and angles within any small area are essentially true. Graticule spacing increases away from central meridian. Equator is straight. Other parallels are complex curves concave toward nearest pole. Central meridian and each meridian 90° from it are straight. Other meridians are complex curves concave toward central meridian. [15]

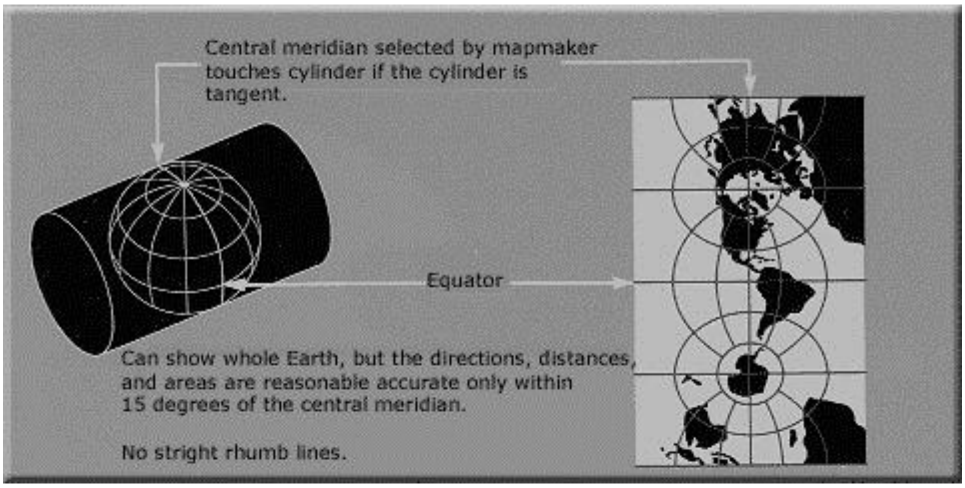

Figure (2.16): Transverse Mercator [20]

The formulas to derive the projected Easting and Northing coordinates are in the form of a series as follow:

Easting:

$$E = FE + k_o v\left[A + (1 - T + C)\frac{A^3}{6} + \left(5 - 18T + T^2 + 72C - 8e^{`2}\right)\frac{A^5}{120}\right] \quad (2.21)$$

Northing:

$$N = FN + k_o\left[M - M_o + v\tan\varphi\left[\begin{matrix}\frac{A^2}{2} + (5 - T + 9C + 4C^2)\frac{A^4}{24} \\ +(61 - 58T + T^2 + 600C - 330e^{`2})\frac{A^6}{720}\end{matrix}\right]\right]$$

(2.22)

Where;

E: Easting

N: Northing

FE: False easting

FN: False northing

λ : Longitude

φ: Latitude

k: Scale factor

$T = \tan^2\varphi$ (2.23)

$C = \frac{e^2}{1-e^2}\cos^2\varphi = e^{`2}\cos^2\varphi$ (2.24)

$A = (\lambda - \lambda_o)\cos\varphi$, with λ and λ_o in radians (2.25)

M: Arc meridian

$$M = a.\begin{bmatrix} \left(1 - \frac{e^2}{4} - \frac{3e^4}{64} - \frac{65e^6}{256} - \cdots\right)\varphi \\ -\left(\frac{3e^2}{8} + \frac{3e^4}{32} + \frac{45e^6}{1024} + \cdots\right)\sin 2\varphi \\ +\left(\frac{15e^4}{256} + \frac{45e^6}{1024} + \cdots\right)\sin 4\varphi \\ -\left(\frac{35e^6}{3072} + \cdots\right)\sin 6\varphi + \cdots \end{bmatrix} \tag{2.26}$$

v: Radius of curvature in the prime vertical

ρ: Radius of curvature in the meridian

With φ in radians and M for φ, the latitude of the origin, derived in the same way. [5]

The reverse formulas to convert Easting and Northing projected coordinates to latitude and longitude are:

$$\varphi = \varphi_1 - \frac{vl\tan\varphi_1}{\rho_1}\begin{bmatrix} \frac{D^2}{2} - \left(5 - 3T_1 + 10C_1 - 4C_1{}^2 - 9e^{`2}\right)\frac{D^4}{24} \\ + \frac{\left(61 + 90T_1 + 298C_1 + 45T^2 - 252e^{`2} - 3C_1{}^2\right)D^6}{720} \end{bmatrix} \tag{2.27}$$

$$\lambda = \lambda_0 + \begin{bmatrix} D - (1 + 2T_1 + C_1)\frac{D^3}{6} \\ +(5 - 2C_1 + 28T_1 - 3C_1{}^2 + 8e^{`2} + 24T_1{}^2)\frac{D^5}{120} \end{bmatrix} / \cos\varphi_1 \tag{2.28}$$

Where φ_1 may be found as for Cassini projection from:

$$v_1 = \frac{a}{\sqrt{1 - e^2\sin^2\varphi^1}} \tag{2.29}$$

$$\rho_1 = \frac{a(1-e^2)}{(1-e^2\sin^2\varphi_1)^{\frac{3}{2}}} \tag{2.30}$$

$$\varphi_1 = \mu_1 + \left(\frac{3e_1}{2} - \frac{27e_1{}^3}{32} + \cdots\right)\sin 2\mu_1 + \left(\frac{21e_1{}^2}{16} - \frac{55e_1{}^4}{32} + \cdots\right)\sin 4\mu_1 + \left(\frac{151e_1{}^3}{96} + \cdots\right)\sin 6\mu_1 + \left(\frac{1097e_1{}^4}{512} - \cdots\right)\sin 8\pi_1 + \cdots \tag{2.31}$$

And where:

$$e_1 = \frac{1-(1-e^2)^{1/2}}{1+(1-e^2)^{1/2}} \tag{2.32}$$

$$\mu_1 = \frac{M_1}{a.(1-\frac{e^2}{4}-\frac{3e^4}{64}-\frac{5e^6}{256}-\cdots)} \quad (2.33)$$

$$M_1 = M_o + (N - FN)/k_o \quad (2.34)$$

$$T_1 = \tan^2\varphi_1 \quad (2.35)$$

$$C_1 = \frac{e^2}{1-e^2}\cos^2\varphi = e^{`2}\cos^2\varphi \quad (2.36)$$

$$e^{`2} = e^2/[1 - e^2] \quad (2.37)$$

$$D = (E - FE) / (v_1 k_0) \quad (2.38)$$

To define coordinates system using Transverse Mercator, the following parameters have to be defined reference ellipsoid

- False Easting
- False Northing
- Central Meridian
- Scale Factor
- Latitude of Origin
- Scale Factor at Central Meridian

In Palestine, there is a coordinates system named Palestinian Transverse Mercator or Palestine-1923-Belt with the following parameters: [5]

• False Easting	170251.555000
• False Northing	1126867.909000
• Central Meridian	35.212081
• Scale Factor	1.000000
• Latitude of Origin	31.734097
• Spheroid	Clarke_1880_Benoit
• Semi major axis	6378300.790000000
• Semi minor axis	6356566.430000036
• Inverse flattening	293.46623457099997

The other common system is the Israel Transverse Mercator (Israel-TM-Grid) with the following parameters: [5]

• False Easting	219529.584000
• False Northing	626907.39000
• Central Meridian	35.204517
• Scale Factor	1.000007
• Latitude of Origin	31.734394
• Spheroid	GRS1980
• Semi major axis	6378137.000000
• Semi minor axis	6356752.314000
• Inverse flattening	298.2572221

2.4.3. Cassini_Soldner Projection

The name Cassini-Soldner refers to the more accurate ellipsoidal version, developed in the 19th century. This transverse cylindrical projection maintains scale along the central meridian and all lines parallel to it and is neither equal area nor conformal. It is most suited for large scale mapping of areas predominantly north-south in extent. [17]

The formula to derive projected Easting and Northing coordinates are:

Easting:

$$E = FE + v[A - T.A^3/6 - (8 - T + 8C)T.A^3/120 \quad (2.39)$$

Northing:

$$N = FN + M - M_o + v.tan\varphi[A^2/2 + (5 - T + 6C)A^4/24] \quad (2.40)$$

Where;

E: Easting

N: Northing

FE: False easting

FN: False northing

λ : Longitude on the central meridian

λ : Longitude

φ: Latitude

$$T = \tan^2\varphi \quad (2.41)$$

$$C = \frac{e^2}{1-e^2}\cos^2\varphi = e'^2\cos^2\varphi \quad (2.42)$$

$$A = (\lambda - \lambda_o)\cos\varphi \qquad \text{,with } \lambda \text{ and } \lambda_o \text{ in radians} \quad (2.43)$$

And M, the distance along the meridian from equator to latitude Ø (Arc Meridian), is given by:

$$M = a.\begin{bmatrix} \left(1 - \frac{e^2}{4} - \frac{3e^4}{64} - \frac{65e^6}{256} - \cdots\right)\varphi \\ -\left(\frac{3e^2}{8} + \frac{3e^4}{32} + \frac{45e^6}{1024} + \cdots\right)\sin 2\varphi \\ +\left(\frac{15e^4}{256} + \frac{45e^6}{1024} + \cdots\right)\sin 4\varphi \\ -\left(\frac{35e^6}{3072} + \cdots\right)\sin 6\varphi + \cdots \end{bmatrix} \quad (2.44)$$

With ∅ in radians, M_o is the value of M calculated for latitude of the chosen origin. This may not necessarily be chosen as the equator.

To compute latitude and longitude from Easting and Northing, the reverse formulas are:

$$\varphi = \varphi_1 - \frac{v \tan\varphi^1}{\rho 1}\left[\frac{D^2}{2} - (1 + 3T^1)\frac{D^4}{24}\right] \tag{2.45}$$

$$\lambda = \lambda_o + \frac{\left[D - T^1 \frac{D^3}{3} + (1 + 3T^1) + \frac{T^1 D^5}{15}\right]}{\cos\varphi^1} \tag{2.46}$$

where ρ_1 is ρ calculated at $\varphi = \varphi_1$, and φ_1 is the latitude of the point on the central meridian which has the same northing as the point whose coordinates are sought and is found from: [5]

$$v_1 = \frac{a}{\sqrt{1 - e^2 sin^2 \emptyset_1}} \tag{2.47}$$

$$\rho^1 = \frac{a(1 - e^2)}{(1 - e^2 \sin^2\varphi_1)^{\frac{3}{2}}} \tag{2.48}$$

$$\varphi_1 = \mu^1 + \left(\frac{3e^1}{2} - \frac{27e^{1^3}}{32} + \cdots\right)\sin 2\mu^1 + \left(\frac{21e^{1^2}}{16} - \frac{55e^{1^4}}{32} + \cdots\right)\sin 4\mu^1$$

$$+ \left(\frac{151e_1{}^3}{96} + \cdots\right)\sin 6\mu_1 + \left(\frac{1097e_1{}^4}{512} - \cdots\right)\sin 8\pi_1 + \cdots \tag{2.49}$$

Where;

$$e^1 = \frac{1 - (1 - e^2)^{\frac{1}{2}}}{1 + (1 - e^2)^{\frac{1}{2}}} \tag{2.50}$$

$$\mu_1 = \frac{M^1}{a.\left(1 - \frac{e^2}{4} - \frac{3e^4}{64} - \frac{5e^6}{256} - \cdots\right)} \tag{2.51}$$

$$M_1 = M_o + (N - FN) \tag{2.52}$$

$$T_1 = \tan^2\varphi_1 \tag{2.53}$$

$$D = (E - FE)/v_1 \tag{2.54}$$

To define a coordinate system using Cassini projection the following parameters are to be considered reference ellipsoid:

- False Easting
- False Northing
- Central Meridian
- Scale Factor = 1
- Latitude of Origin

The Palestinian grid named Palestine_1923_Grid is built using Cassini projection, which normally used in land surveying and engineering projects with the following parameters: [5]

• False Easting	170251.555000
• False Northing	126867.909000
• Central Meridian	35.212081
• Scale Factor	1.000000
• Latitude of Origin	31.734.097
• Spheroid	Clarke_1880_Benoit
• Semi major axis	6378300.790000000
• Semi minor axis	6356566.430000036
• Inverse flattening	293.46623457099997

Israel Old Grid is the same of Palestine grid (Paestine-1923-Grid), but 1 million is added to the northing value: [5]

• False Easting	170251.555000
• False Northing	1126867.909000
• Central Meridian	35.212081
• Scale Factor	1.000000
• Latitude of Origin	31.734.097
• Spheroid	Clarke_1880_Benoit
• Semi major axis	6378300.790000000
• Semi minor axis	6356566.430000036
• Inverse flattening	293.46623457099997

2.4.5 Universal Transverse Mercator (UTM)

UTM coordinate system is used in survey navigation and in GIS and it's the most commonly used in Transverse Mercator. From the figure (2.17) below, we can see that UTM zone numbers designate 6 degrees longitudinal strips extending from 80 degree south latitude to 84 degree north longitude. [5]

To find the central meridian of a UTM zone: [5]

$$\text{Central Meridian} = (\text{zone\#} * 6 - 3) - 180 \qquad (2.55)$$

To find which zone you belong to at a given longitude: [5]

$$\text{Zone} = \text{int}\left\{\frac{\lambda+180}{6}\right\} + 1 \qquad (2.56)$$

Example (2.1):

Find the central meridian of zone 36:

Central meridian = (36 * 6 – 3) – 180 = 33° east.

Example (2.2):

Find the zone that belong to 33° east longitude:

Zone = int { (33 + 180) / 6 } + 1 = int { 35.5 } + 1 = 35 + 1 = 36

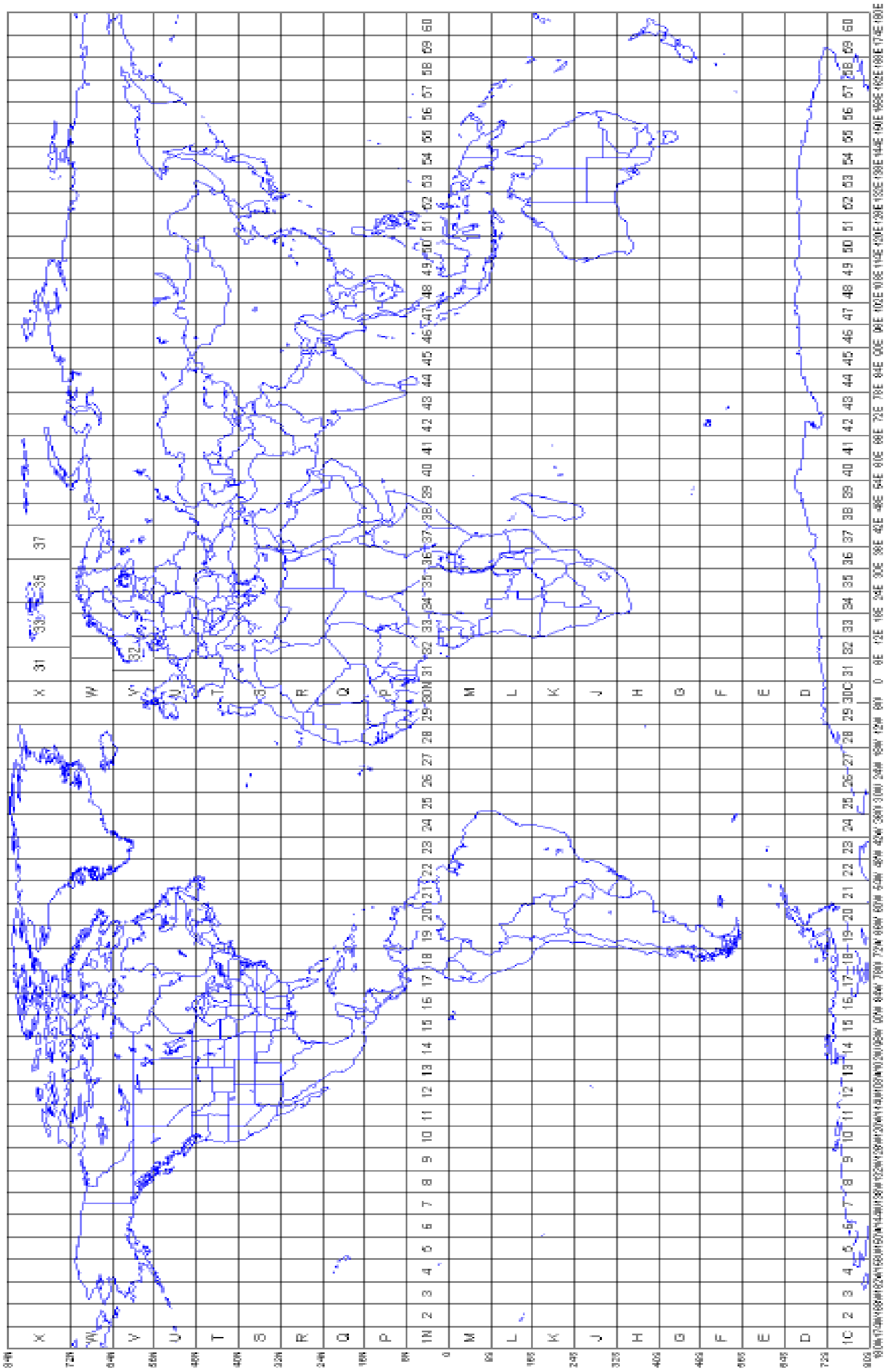

Figure (2.17): Universal Transverse Mercator [12]

CHAPTER 3

DATUM TRANSFORMATION

3.1. Introduction

3.2. Datum & Ellipsoids

3.3. Datum Transformations

3.4. Precision & Blunder detection.

3.1. Introduction

The coordinates of all locations on the earth are defined referring to a datum. While a spheroid nearly represents the shape of the earth, a datum defines the position of a spheroid relative to the center of the earth. A point on the surface of the earth is matched to a particular position on the surface of the ellipsoid. This point is known as the origin point of the coordinates system on the datum. The coordinates of the origin point coordinates system are fixed, and all other points are calculated referring to it. The coordinate system origin of a local datum is not at the center of the earth. The center of the spheroid local datum is offset from the earth's center, depending on a global datum like WGS 84. [5]

A datum provides a frame or reference for measuring locations on the surface of the earth. It defines the origin and orientation of latitude and longitude lines. Whenever change the datum, or more correctly, the geographic coordinate system, the coordinate values of a point will change. [5]

3.2. Datum and Ellipsoids

The shape of the earth is ellipsoid because the distance from the center of the earth to the equator is larger than the distance from the center to the poles by about 23km.

To make an ellipsoid model of the earth, rotate the ellipse about the shorter polar axis (semi-minor axis b) to form a solid surface, see figure (3.1) a datum is defined by choosing an ellipsoid and then a primary reference point.

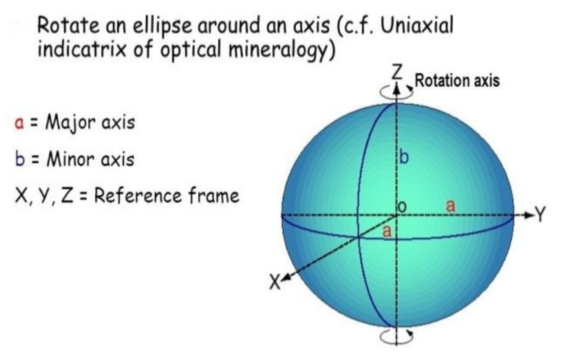

Figure (3.1): Ellipse Rotation [7]

The reference ellipsoid of the Palestine_1923_Grid, Palestine_1923_Israel_CS_Grid and Palestine_1923_Belt is the Clarke_1880_Benoit. The reference ellipsoid of the Israel_TM_Grid is the Geodetic Reference System of 1980 (GRS80). [5]

3.3. Datum Transformation

3.3.1. Introduction

Equation-based transformation methods can classified into the following basic four methods. Having data in one datum and needing the coordinates in other is a common task in geodesy, surveying and GPS. A transformation must be used to display coordinates from a GPS receiver in any other datum than WGS84. Over any small area (150x150 km), the transformation will be a constant shift in latitude, longitude and height, neglecting scaling and rotation. But for countries with large areas seven parameter are preferred; scale, three rotations around the three axis and three translations.

There are three common methods of making these transformations from one datum to another. These methods are 3D similarity, Helmert and Molodensky method, Figure (3.2) shows the basic parameters for datum transformation.

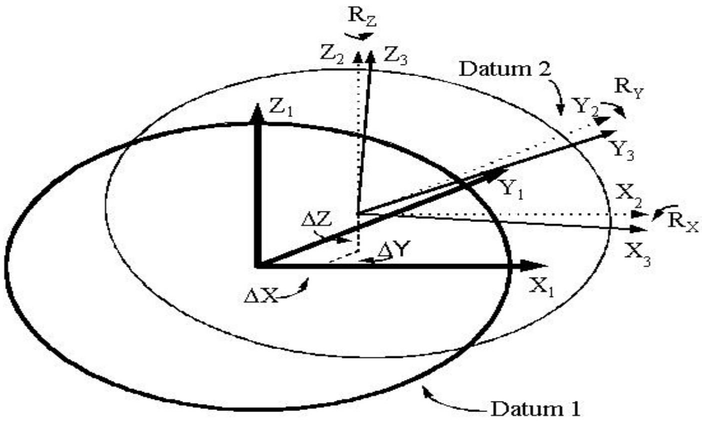

Figure (3.2): Datum Transformation [6]

3.3.2. Least square solution

To solve the datum transformation problem, control points with known positions in both datums have to be available. To solve the parameters, nonlinear least square solution method is used with basic matrix equation. For (m) number of observation and (n) number of unknowns.

$$AX = L+V \tag{3.1}$$

Where A is the matrix of coefficients multiplied by the unknown parameters:

$$A = \begin{bmatrix} a_{11} & a_{12} & \Lambda & a_{1n} \\ a_{21} & a_{22} & \Lambda & a_{2n} \\ M & M & M & M \\ a_{m1} & a_{m2} & \Lambda & a_{mn} \end{bmatrix} \tag{3.2}$$

And X is the matrix of unknowns:

$$X = \begin{bmatrix} dx_1 \\ dx_2 \\ M \\ dx_n \end{bmatrix} \tag{3.3}$$

And L is observations matrix, which is the difference between the measured values and the computed ones using the initial values of the unknown parameters:

$$L = \begin{bmatrix} l_1 - l_{1o} \\ l_2 - l_{2o} \\ M \\ l_m - l_{mo} \end{bmatrix} \tag{3.4}$$

V is the residuals matrix:

$$V = \begin{bmatrix} v_1 \\ v_2 \\ M \\ vm \end{bmatrix} \tag{3.5}$$

Multiply (AX = L + V) on left by (A^T) to get:

$$A^T AX = A^T L \tag{3.6}$$

Where;

$$A^T A = \begin{bmatrix} a_{11} & a_{21} & \Lambda & a_{m1} \\ a_{12} & a_{22} & \Lambda & a_{m2} \\ M & M & M & M \\ a_{1n} & a_{2n} & \Lambda & a_{mn} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \Lambda & a_{1n} \\ a_{21} & a_{22} & \Lambda & a_{2n} \\ M & M & M & M \\ a_{m1} & a_{m2} & \Lambda & a_{mn} \end{bmatrix} = N \tag{3.7}$$

$$\begin{array}{llll} n_{11} = \sum_{i=1}^{m} a_{i1}^2 & n_{12} = \sum_{i=1}^{m} a_{i1}a_{i2} & \Lambda & n_{1n} = \sum_{i=1}^{m} a_{i1}a_{in} \\ n_{21} = \sum_{i=1}^{m} a_{i2}a_{i1} & n_{22} = \sum_{i=1}^{m} a_{i2}^2 & \Lambda & n_{2n} = \sum_{i=1}^{m} a_{i2}a_{in} \\ M & M & M & M \\ n_{n1} = \sum_{i=1}^{m} a_{in}a_{i1} & n_{n2} = \sum_{i=1}^{m} a_{in}a_{i2} & \Lambda & n_{nn} = \sum_{i=1}^{m} a_{in}^2 \end{array} \tag{3.8}$$

When observations are not weighted, then:

$$X = (A^T A)^{-1} A^T L \tag{3.9}$$

When weighted observation are used, then:

$$X = (A^T W A)^{-1} A^T W L \tag{3.10}$$

3.3.5. 3D similarity

The simplest datum transformation method is three-parameter transformation or a geocentric translation. When the minor axes of the ellipsoids are assumed to be parallel, to transform assuming that rotations are zeroes, then shifts (T_x, T_y, T_z) are defined from source geocentric coordinate system to target geocentric coordinate system or from local datum to WGS 1984 or another geocentric datum as common in GPS. The three parameters are linear shifts and are always in distance units, and usually called datum shifts.

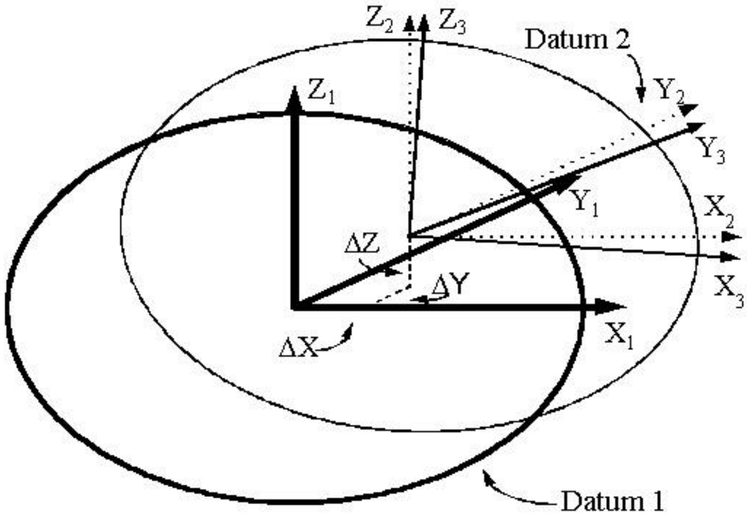

Figure (3.3): 3-Parameter Transformation [6]

The mathematical model for the 3D similarity three-parameter transformation is:

$$X = x + T_x \tag{3.11}$$
$$Y = y + T_y \tag{3.12}$$
$$Z = z + T_z \tag{3.13}$$

In matrix form;

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{3.14}$$

This equation can be obtained from 3D similarity seven-parameter transformation

$$X = S(m_{11}x + m_{21}y + m_{31}z) + T_x \tag{3.15}$$
$$Y = S(m_{12}x + m_{22}y + m_{32}z) + T_y \tag{3.16}$$
$$Z = S(m_{13}x + m_{23}y + m_{33}z) + T_z \tag{3.17}$$

When;

$$S = 1 \tag{3.18}$$

$$\omega = \emptyset = \kappa = 0 \tag{3.19}$$

The least squares solution as linear solution:

$$AX = L + V \tag{3.20}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{3.21}$$

The final solution of the unknowns i:

$$X = (A^T A)^{-1} A^T L \tag{3.22}$$

$$X = \begin{bmatrix} dT_x \\ dT_y \\ dT_z \end{bmatrix} \tag{3.23}$$

The 3-parameters are calculated in equation (3.10), after first iteration, initial values of second iterations are calculated using the final results of the first iteration:

$$T_x = T_{x0} + dT_x \tag{3.24}$$
$$T_y = T_{y0} + dT_y \tag{3.25}$$
$$T_z = T_{z0} + dT_z \tag{3.26}$$

In Palestine_1923_Grid, the transformation is defined by three translations. The values of the translation are taken from Trimble geomatic office software. According to the following equations:

$$X_{Palestine\_1923} = X_{WGS84} + \triangle X \tag{3.27}$$
$$Y_{Palestine\_1923} = Y_{WGS84} + \triangle Y \tag{3.28}$$
$$Z_{Palestine\_1923} = Z_{WGS84} + \triangle Z \tag{3.29}$$

Where;

$$\triangle X = 230.00m, \quad \triangle Y = 71.00m, \quad \triangle Z = -273.00m \tag{3.30}$$

3.3.6. Helmert Transformations

When dealing with map, we have maps with different scales, orientations and coordinates origins. Then coordinates transformations are used to move the coordinates from one map system to the other. These transformations are; 2D Conformal, 2D Affine, 3D Conformal and 3D Linearized. [4]

3.3.6.1. 2D Conformal

In this type of coordinate's transformations as shown in the figure below, we have three steps; scale change, rotation and two translations:

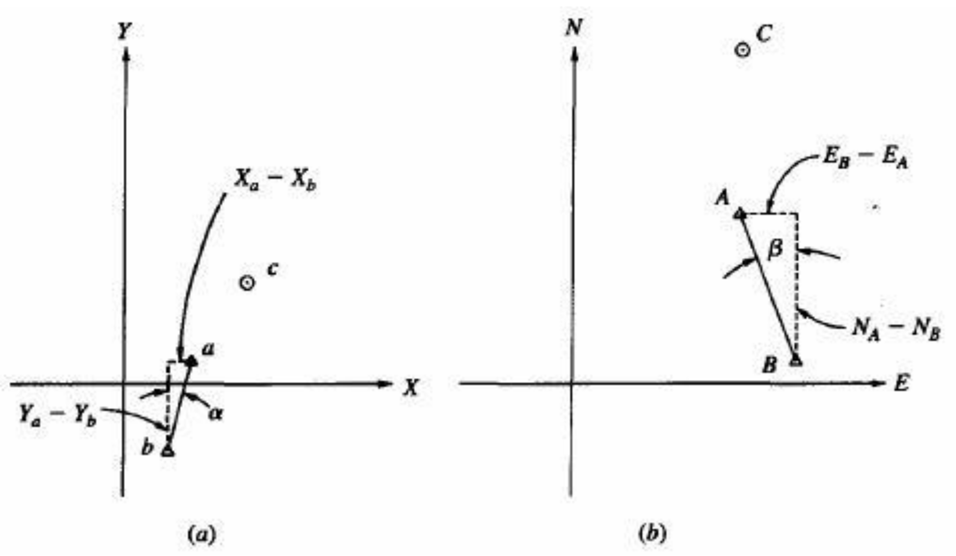

Figure (3.4): 2D Conformal Helmert Transformation [4]

Step 1: Scale Change

• The lengths of lines ab and AB are unequal; hence the scales of the two coordinate systems are unequal.

• The scale of the XY system is made equal to that of the EN system by multiplying each X and Y coordinate by a scale factor s. The scaled coordinates are designated as X' and Y'.

• By use of the two control points, the scale factor is calculated in relation to the two lengths AB and ab as:

$$s = \frac{AB}{ab} = \frac{\sqrt{(E_B - E_A)^2 + (N_B - N_A)^2}}{\sqrt{(X_b - X_a)^2 + (Y_b - Y_a)^2}} \quad (3.31)$$

Step 2: Rotation

• If the scaled X'Y' coordinate system is superimposed over the EN system , so that line ***AB*** in both systems coincide, the result is as shown in Figure below.

• A n auxiliary axis system E'N' is constructed through the origin of the X'Y' axis system parallel to the EN axes.

• I t is necessary to rotate from the X' Y' system to the E'N' system, or in other words, to calculate E'N' coordinates for the unknown points from their X' Y' coordinates.

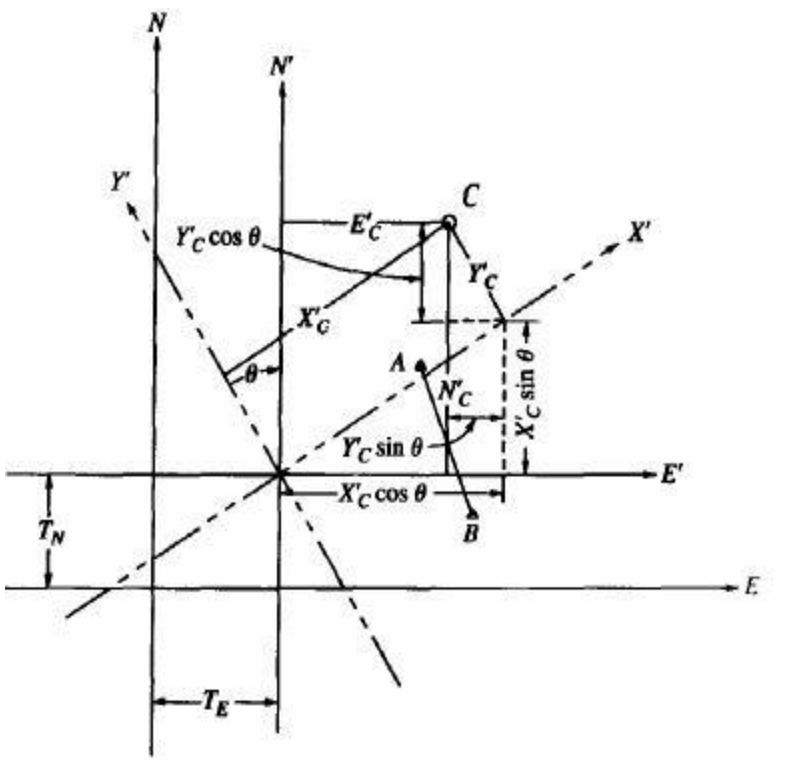

Figure (3.5): Rotation calculation in 2D Conformal [4]

- The E'N' coordinates of point C may be calculated in term of the clockwise angle θ by using the following equations:

$$E'_C = X'_C . \cos\theta - Y'_C . \sin\theta \quad (3.32)$$

$$N'_C = X'_C . \sin\theta + Y'_C . \cos\theta \quad (3.33)$$

Where,

$$\theta = \alpha + \beta \quad (3.34)$$

$$\alpha = tan^{-1}\left(\frac{X_a - X_b}{Y_a - Y_b}\right) \quad (3.35)$$

$$\beta = tan^{-1}\left(\frac{E_B - E_A}{N_B - N_A}\right) \quad (3.36)$$

Or more general:

$$\theta = AZ_{ab} + AZ_{AB} \quad (3.37)$$

- The final step in the coordinate transformation is a translation of the origin of the E'N' system to the origin of the EN system.

- The translation factors required are T_E and T_N, which are illustrated in the above figure. Final E and N ground coordinates for points C then are:

$$E_C = E'_C + T_E \quad (3.38)$$
$$N_C = N'_C + T_N \quad (3.39)$$
$$T_E = E_A - E'_A = E_B - E'_B \quad (3.40)$$
$$T_N = N_A - N'_A = N_B - N'_B \quad (3.41)$$

<u>Other method to write the equations:</u>

For the following coordinates transformations equation:

$$E_a = sX_a \cos\theta - sY_a \sin\theta + T_e \quad (3.42)$$
$$N_a = sX_a \sin\theta + sY_a \cos\theta + T_n \quad (3.43)$$
$$E_b = sX_b \cos\theta - sY_b \sin\theta + T_e \quad (3.44)$$
$$N_b = sX_b \sin\theta + sY_a \cos\theta + T_n \quad (3.45)$$

If we suppose $a = s\cos\theta$ and $b = s\sin\theta$, then:

$$\mathrm{E_A = aX_a - bY_a + T_e} \quad (3.46)$$
$$N_A = aY_a + bX_a + T_n \quad (3.47)$$
$$\mathrm{E_B = aX_b - bY_b + T_e} \quad (3.48)$$
$$N_B = aY_b + bX_b + T_n \quad (3.49)$$
$$s = \sqrt{a^2 + b^2} \quad (3.50)$$
$$\theta = tan^{-1}(\mathrm{b/a}) \quad (3.51)$$

In matrix form:

$$\begin{bmatrix} y \\ x \end{bmatrix}_{Target} = S.R(\theta).\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} = S.\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} \quad (3.52)$$

Where,

x, $\mathrm{y_{Target}}$: the target coordinate system.
x, $\mathrm{y_{Source}}$: the source coordinate system.
S: scale
R(θ): rotation matrix around x,y axis
T_x, T_y: translations

3.3.6.2. 2D Affine

In the affine coordinates transformations, to transform from xy-coordinates to XY- coordinates. As shown in the figure below, we have:

- Different scales in the x direction and y-direction.
- Rotation angle.
- The x-axis and y-axis are not orthogonal.
- Two translations in the x and y directions.

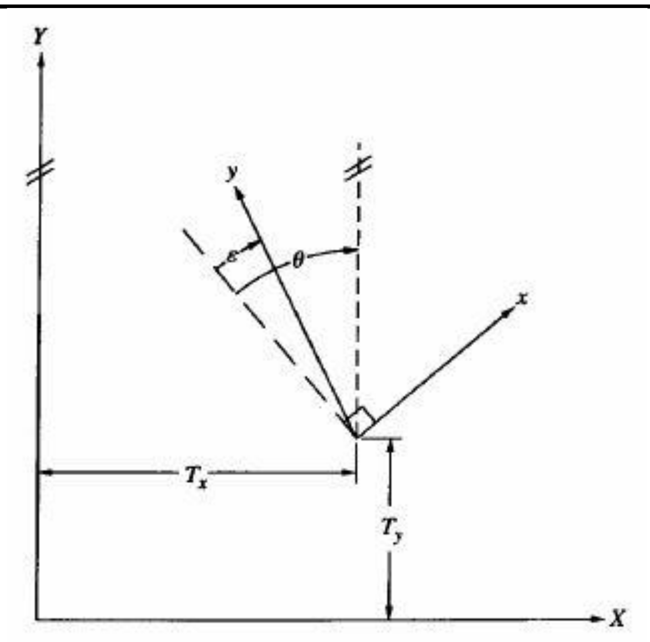

Figure (3.6): 2D Affine Helmert Transformation [4]

Step 1: scaling in the x and y direction

If we scale the xy-coordinates we get new coordinates system y'y'-coordinates:

$$x' = s_x x \tag{3.53}$$

$$y' = s_y y \tag{3.54}$$

Step 2: Non-Orthogonality Correction

this step we find new coordinates system x"y". so that x"-axis and y"-axis are orthogonal. In the figure below, ε is the non-orthogonality angle.

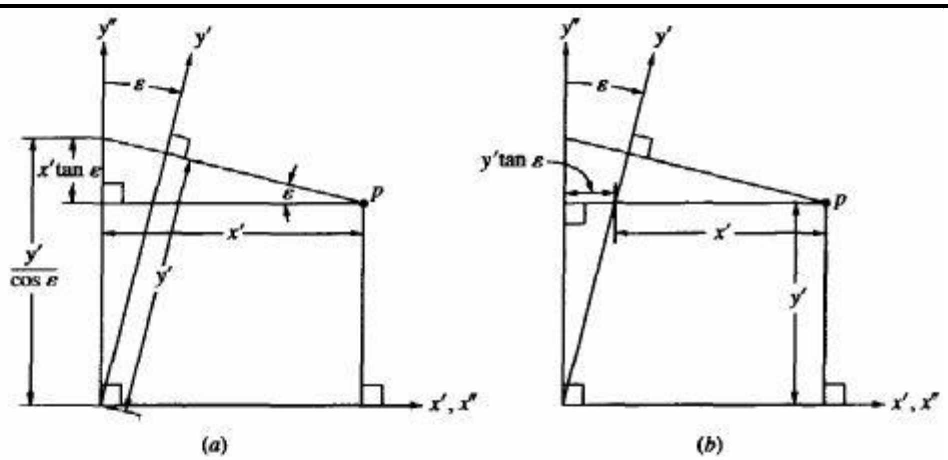

Figure (3.7): 2D Affine Non-Orthogonality Correction [4]

$$x" = x' \tag{3.55}$$

$$y" = y'/cos\,\varepsilon \; - x' tan\,\varepsilon \tag{3.56}$$

Step 3: Rotation:

In this step we rotate the x”y”-coordinates X’Y’-coordinates, where X’Y’- coordinate are parallel to XY-coordinates. θ is the rotation angle between the y”-axis and the Y’-axis:

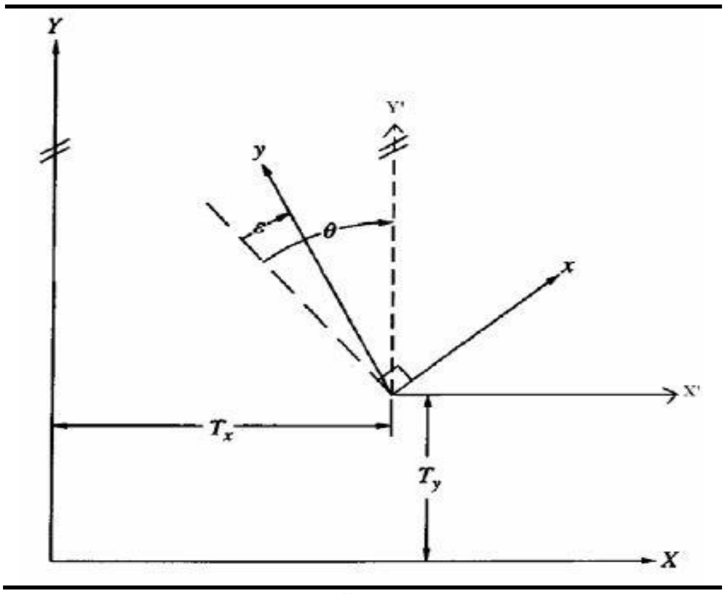

Figure (3.8): 2D Affine Rotation Angle [4]

$$X' = x\cos\theta - y\sin\theta \tag{3.57}$$
$$Y' = x\sin\theta + y\cos\theta \tag{3.58}$$

Step 4: Translation:

In this step we add shift values to X’Y’-coordinates to get XY required coordinates:

$$X = X' + T_x \tag{3.59}$$
$$Y = Y' + T_y \tag{3.60}$$

Using the values of X’Y’, we get:

$$X = s_x x\cos\theta - \left(\frac{s_y y}{\cos\varepsilon} - s_x x\sin\varepsilon\right)\sin\theta + T_x \tag{3.61}$$
$$Y = s_x x\sin\theta - \left(\frac{s_y y}{\cos\varepsilon} - s_x x\tan\varepsilon\right)\tan\theta + T_y \tag{3.62}$$

This can be arranged as follows:

$$X = s_x x(\cos\theta + \tan\varepsilon\sin\theta) - s_y y\frac{\sin\theta}{\cos\varepsilon} + T_X \tag{3.63}$$
$$Y = s_x x(\sin\theta + \tan\varepsilon\cos\theta) - s_y y\frac{\cos\theta}{\cos\varepsilon} + T_Y \tag{3.64}$$

In other way, we arrange the above equations as follows:

$$X = T_X + s_x x\frac{\cos\varepsilon\cos\theta + \sin\varepsilon\sin\theta}{\cos\varepsilon} - s_y y\frac{\sin\theta}{\cos\varepsilon} \tag{3.65}$$
$$Y = T_Y + s_x x\frac{\cos\varepsilon\sin\theta + \sin\varepsilon\cos\theta}{\cos\varepsilon} + s_y y\frac{\cos\theta}{\cos\varepsilon} \tag{3.66}$$

Using the trigonometric function relation:

$$X = T_X + s_x x \frac{\cos(\varepsilon-\theta)}{\cos\varepsilon} - s_y y \frac{\sin\theta}{\cos\varepsilon} \tag{3.67}$$

$$Y = T_Y + s_x x \frac{\sin(\varepsilon-\theta)}{\cos\varepsilon} - s_y y \frac{\cos\theta}{\cos\varepsilon} \tag{3.68}$$

Other form of the above equation is:

$$X = a_0 + a_1 x + a_2 y \tag{3.69}$$

$$Y = b_0 + b_1 x + b_2 y \tag{3.70}$$

Where,

$$a_0 = T_X \tag{3.71}$$

$$a_1 = s_x x \frac{\cos(\varepsilon-\theta)}{\cos\varepsilon} \tag{3.72}$$

$$a_2 = s_y y \frac{\sin\theta}{\cos\varepsilon} \tag{3.73}$$

$$b_0 = T_Y \tag{3.74}$$

$$b_1 = s_x x \frac{\sin(\varepsilon-\theta)}{\cos\varepsilon} \tag{3.75}$$

$$b_2 = s_y y \frac{\cos\theta}{\cos\varepsilon} \tag{3.76}$$

In the opposite direction we get:

$$\theta = \tan^{-1}\left(\frac{-a_2}{b_2}\right) \tag{3.77}$$

$$\varepsilon - \theta = \tan^{-1}\left(\frac{b_1}{a_1}\right) \tag{3.78}$$

$$s_x = a_1 \frac{\cos\varepsilon}{\cos(\epsilon-\theta)} \tag{3.79}$$

$$s_y = b_2 \frac{\cos\varepsilon}{\cos\theta} \tag{3.80}$$

$$T_X = a_0 \tag{3.81}$$

$$T_Y = b_0 \tag{3.82}$$

In matrix form:

$$\begin{bmatrix} y \\ x \end{bmatrix}_{Target} = A.\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} \tag{3.83}$$

$$\begin{bmatrix} y \\ x \end{bmatrix}_{Target} = R.V.\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.\begin{bmatrix} E_{yy} & E_{yx} \\ E_{xy} & E_{xx} \end{bmatrix}.\begin{bmatrix} y \\ x \end{bmatrix}_{Source} + \begin{bmatrix} T_y \\ T_x \end{bmatrix} \tag{3.84}$$

Where,

x, y_{Target} : the target coordinate system.

x, y_{Source} : the source coordinate system.

R: rotation matrix around x,y axis.

V: distortion matrix.

T_x, T_y: translations.

3.3.6.3. 3D Conformal

The 3D Conformal Helmert Transformation has seven parameter transformations that include the three translation parameters, three rotation parameters and a scale parameter

Parameters are: S, ω, ∅, k, T_x, T_y and T_z the equations for the 3D Conformal Helmert transformation are:

$$X = S(m_{11}x + m_{21}y + m_{31}z) + T_x \tag{3.85}$$
$$Y = S(m_{12}x + m_{22}y + m_{32}z) + T_y \tag{3.86}$$
$$Z = S(m_{13}x + m_{23}y + m_{33}z) + T_z \tag{3.87}$$

In matrix form:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = S \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{3.88}$$

$$X = S.M.x + T_x \tag{3.89}$$
$$Y = S.M.y + T_y \tag{3.90}$$
$$Z = S.M.z + T_z \tag{3.91}$$

Where;

X, Y, Z: Coordinate system in the first datum
x, y, z: Coordinate system in the second datum
S: Scale
M: Rotation matrix
T_x, T_y, T_z: Translation matrix

$$m_{11} = \cos\emptyset \cos\kappa \tag{3.92}$$
$$m_{12} = \sin\omega \sin\emptyset \cos\kappa + \cos\omega \sin\kappa \tag{3.93}$$
$$m_{13} = -\cos\omega \sin\emptyset \cos\kappa + \sin\omega \sin\kappa \tag{3.94}$$
$$m_{21} = -\cos\emptyset \sin\kappa \tag{3.95}$$
$$m_{22} = -\sin\omega \sin\emptyset \sin\kappa + \cos\omega \cos\kappa \tag{3.96}$$
$$m_{23} = \cos\omega \sin\emptyset \sin\kappa + \sin\omega \cos\kappa \tag{3.97}$$
$$m_{31} = \sin\emptyset \tag{3.98}$$
$$m_{32} = -\sin\omega \cos\emptyset \tag{3.99}$$
$$m_{33} = \cos\omega \cos\emptyset \tag{3.100}$$

In the system of the equations, seven-parameter require a minimum number of two horizontal control stations with known X-Y and x-y-z coordinates, in addition to three stations with known Z and x-y-z coordinates. If there is more than the minimum number of observation, a least-square solution can be applied. The equations are nonlinear with respect to the unknowns and must be linearized for a solution. The following linearized equations for a full control point (Horizontal and Vertical). [4]

$$\begin{bmatrix} (\frac{\partial X}{\partial S})_0 & 0 & (\frac{\partial X}{\partial \emptyset})_0 & (\frac{\partial X}{\partial \kappa})_0 & 1 & 0 & 0 \\ (\frac{\partial Y}{\partial S})_0 & (\frac{\partial Y}{\partial \omega})_0 & (\frac{\partial Y}{\partial \emptyset})_0 & (\frac{\partial Y}{\partial \kappa})_0 & 0 & 1 & 0 \\ (\frac{\partial Z}{\partial S})_0 & (\frac{\partial Z}{\partial \omega})_0 & (\frac{\partial Z}{\partial \emptyset})_0 & (\frac{\partial Z}{\partial \kappa})_0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dS \\ d\omega \\ d\emptyset \\ d\kappa \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad (3.101)$$

Where;

$$\frac{\partial X}{\partial S} = m_{11}x + m_{21}y + m_{31}z \quad (3.102)$$

$$\frac{\partial Y}{\partial S} = m_{12}x + m_{22}y + m_{32}z \quad (3.103)$$

$$\frac{\partial Z}{\partial S} = m_{13}x + m_{23}y + m_{33}z \quad (3.104)$$

$$\frac{\partial Y}{\partial \omega} = -S(m_{13}x + m_{23}y + m_{33}z) \quad (3.105)$$

$$\frac{\partial Z}{\partial \omega} = S(m_{12}x + m_{22}y + m_{32}z) \quad (3.106)$$

$$\frac{\partial X}{\partial \emptyset} = S[-\sin(\emptyset)\cos(\kappa)\,x + \sin(\emptyset)\sin(\kappa)\,y + \cos\emptyset\,z] \quad (3.107)$$

$$\frac{\partial Y}{\partial \emptyset} = S[\sin(\omega)\cos(\emptyset)\cos(\kappa)\,x - \sin(\omega)\cos(\emptyset)\sin(\kappa)\,y + \sin(\omega)\sin(\emptyset)\,z] \quad (3.108)$$

$$\frac{\partial Z}{\partial \emptyset} = S[-\cos(\omega)\cos(\emptyset)\cos(\kappa)\,x + \cos(\omega)\cos(\emptyset)\sin(\kappa)\,y - \cos(\omega)\sin(\emptyset)\,z] \quad (3.109)$$

$$\frac{\partial X}{\partial \kappa} = S(m_{21}x - m_{11}y) \quad (3.110)$$

$$\frac{\partial Y}{\partial \kappa} = S(m_{22}x - m_{12}y) \quad (3.111)$$

$$\frac{\partial Z}{\partial \kappa} = S(m_{23}x - m_{13}y) \quad (3.112)$$

The matrix solution is:

$$X = (A^T A)^{-1} A^T L \quad (3.113)$$

$$X = \begin{bmatrix} dS \\ d\omega \\ d\emptyset \\ d\kappa \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} \quad (3.114)$$

The seven-parameter are calculated after first iteration, an assigned as the initial value of second iteration, where:

$$S = S_0 + dS \quad (3.115)$$

$$\omega = \omega_0 + d\omega \quad (3.116)$$

$$\emptyset = \emptyset_0 + d\emptyset \tag{3.117}$$
$$\kappa = \kappa_0 + d\kappa \tag{3.118}$$
$$T_x = T_{x0} + dT_x \tag{3.119}$$
$$T_y = T_{y0} + dT_y \tag{3.120}$$
$$T_z = T_{z0} + dT_z \tag{3.121}$$

The rotation matrix M for the transformation is:

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \tag{3.122}$$

To calculate the elements of last matrix we must give initial values for the parameters, using one, normally, the following are useful.

$$S_0 = 1 \tag{3.123}$$
$$\omega_0 = \emptyset_0 = \kappa_0 = 0 \tag{3.124}$$
$$T_{x0} = (X - x) \tag{3.125}$$
$$T_{y0} = (Y - y) \tag{3.126}$$
$$T_{z0} = (Z - z) \tag{3.127}$$

To calculate the initial values of the observations:

$$X_0 = S(m_{11}x + m_{21}y + m_{31}z) + T_{x0} \tag{3.128}$$
$$Y_0 = S(m_{12}x + m_{22}y + m_{32}z) + T_{y0} \tag{3.129}$$
$$Z_0 = S(m_{13}x + m_{23}y + m_{33}z) + T_{z0} \tag{3.130}$$

$$m_{11} = \cos\emptyset \cos\kappa \tag{3.131}$$
$$m_{11} = \cos 0 \cos 0 = 1 \tag{3.132}$$

$$m_{12} = \sin\omega \sin\emptyset \cos\kappa + \cos\omega \sin\kappa \tag{3.133}$$
$$m_{12} = \sin 0 \sin 0 \cos 0 + \cos 0 \sin 0 = 0 \tag{3.134}$$

$$m_{13} = -\cos\omega \sin\emptyset \cos\kappa + \sin\omega \sin\kappa \tag{3.135}$$
$$m_{13} = -\cos 0 \sin 0 \cos 0 + \sin 0 \sin 0 = 0 \tag{3.136}$$

$$m_{21} = -\cos\emptyset \sin\kappa \tag{3.137}$$
$$m_{21} = -\cos 0 \sin 0 = 0 \tag{3.138}$$

$$m_{22} = -\sin\omega \sin\emptyset \sin\kappa + \cos\omega \cos\kappa \tag{3.139}$$
$$m_{22} = -\sin 0 \sin 0 \sin 0 + \cos 0 \cos 0 = 1 \tag{3.140}$$

$$m_{23} = \cos\omega \sin\emptyset \sin\kappa + \sin\omega \cos\kappa \tag{3.141}$$
$$m_{23} = \cos 0 \sin 0 \sin 0 + \sin 0 \cos 0 = 0 \tag{3.142}$$

$$m_{31} = \sin\emptyset \tag{3.143}$$
$$m_{31} = \sin 0 = 0 \tag{3.144}$$

$$m_{32} = -\sin\omega\cos\emptyset \tag{3.145}$$
$$m_{32} = -\sin 0\cos 0 = 0 \tag{3.146}$$

$$m_{33} = \cos\omega\cos\emptyset \tag{3.147}$$
$$m_{33} = \cos 0\cos 0 = 1 \tag{3.148}$$

And;

$$\frac{\partial X}{\partial S} = m_{11}x + m_{21}y + m_{31}z = x \tag{3.149}$$

$$\frac{\partial Y}{\partial S} = m_{12}x + m_{22}y + m_{32}z = y \tag{3.150}$$

$$\frac{\partial z}{\partial S} = m_{13}x + m_{23}y + m_{33}z = z \tag{3.151}$$

$$\frac{\partial Y}{\partial \omega} = -S(m_{13}x + m_{23}y + m_{33}z) = -z \tag{3.152}$$

$$\frac{\partial Z}{\partial \omega} = S(m_{12}x + m_{22}y + m_{32}z) = y \tag{3.153}$$

$$\frac{\partial X}{\partial \emptyset} = S[-\sin(\emptyset)\cos(\kappa)\,\mathrm{x} + \sin(\emptyset)\sin(\kappa)\,y + \cos(\emptyset)\,z] = z \tag{3.154}$$

$$\frac{\partial Y}{\partial \emptyset} = S[\sin(\omega)\cos(\emptyset)\cos(\kappa)\,\mathrm{x} - \sin(\omega)\cos(\emptyset)\sin(\kappa)\,y + \sin(\omega)\sin(\emptyset)\,z] = 0 \tag{3.155}$$

$$\frac{\partial Z}{\partial \emptyset} = S[-\cos(\omega)\cos(\emptyset)\cos(\kappa)\,\mathrm{x} + \cos(\omega)\cos(\emptyset)\sin(\kappa)\,y - \cos(\omega)\sin(\emptyset)\,z] = -x \tag{3.156}$$

$$\frac{\partial X}{\partial \kappa} = S(m_{21}x - m_{11}y) = -y \tag{3.157}$$

$$\frac{\partial Y}{\partial \kappa} = S(m_{22}x - m_{12}y) = x \tag{3.158}$$

$$\frac{\partial Z}{\partial \kappa} = -S(m_{23}x - m_{13}y) = 0 \tag{3.159}$$

The linearized equations of each control point, at first iteration one:

$$\begin{bmatrix} (\frac{\partial X}{\partial S})_0 & 0 & (\frac{\partial X}{\partial \emptyset})_0 & (\frac{\partial X}{\partial \kappa})_0 & 1 & 0 & 0 \\ (\frac{\partial Y}{\partial S})_0 & (\frac{\partial Y}{\partial \omega})_0 & (\frac{\partial Y}{\partial \emptyset})_0 & (\frac{\partial Y}{\partial \kappa})_0 & 0 & 1 & 0 \\ (\frac{\partial Z}{\partial S})_0 & (\frac{\partial Z}{\partial \omega})_0 & (\frac{\partial Z}{\partial \emptyset})_0 & (\frac{\partial Z}{\partial \kappa})_0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dS \\ d\omega \\ d\emptyset \\ d\kappa \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{3.160}$$

$$\begin{bmatrix} x & 0 & z & -y & 1 & 0 & 0 \\ y & -z & 0 & x & 0 & 1 & 0 \\ z & y & -x & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dS \\ d\omega \\ d\emptyset \\ d\kappa \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad (3.161)$$

The calculate matrix X and calculate the seven parameter S,ω, $\emptyset$, κ, T_x, T_y, T_z

$$S = S_0 + dS \quad (3.162)$$
$$\omega = \omega_0 + d\omega \quad (3.163)$$
$$\emptyset = \emptyset_0 + d\emptyset \quad (3.164)$$
$$\kappa = \kappa_0 + d\kappa \quad (3.165)$$
$$T_x = T_{x0} + dT_x \quad (3.166)$$
$$T_y = T_{y0} + dT_y \quad (3.167)$$
$$T_z = T_{z0} + dT_z \quad (3.168)$$

3.3.6.4. 3D Linearized

The 3D Linearized Helmert transformations parameters are three linear shifts (T_x, T_y, T_z), three angular rotations around each axis (r_x, r_y, r_z), and scale factor (S). the rotation values are given in decimal seconds.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{new} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + S. \begin{bmatrix} 1 & r_z & -r_y \\ -r_z & 1 & r_x \\ r_y & -r_x & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{original} \quad (3.169)$$

$$X_{new} = S(X + r_z Y - r_y Z) + T_x \quad (3.170)$$
$$Y_{new} = S(-r_z X + Y + r_x Z) + T_y \quad (3.171)$$
$$Z_{new} = S(r_y X - r_x Y + Z) + T_z \quad (3.172)$$

Where;

X, Y, Z: point coordinate in the target system
x, y, z: point coordinate in the source
S: scale factor
r_x, r_y, r_z: angular rotations
T_x, T_y, T_z: linear shifts

A least-squares solution can be applied. The equations are nonlinear with respect to the unknowns and must be linearized for a solution. The following linearized equations for a full control point (Horizontal and Vertical).

$$\begin{bmatrix} (\frac{\partial X}{\partial S})_0 & (\frac{\partial X}{\partial r_x})_0 & (\frac{\partial X}{\partial r_y})_0 & (\frac{\partial X}{\partial r_z})_0 & 1 & 0 & 0 \\ (\frac{\partial Y}{\partial S})_0 & (\frac{\partial Y}{\partial r_x})_0 & (\frac{\partial Y}{\partial r_y})_0 & (\frac{\partial Y}{\partial r_z})_0 & 0 & 1 & 0 \\ (\frac{\partial Z}{\partial S})_0 & (\frac{\partial Z}{\partial r_x})_0 & (\frac{\partial Z}{\partial r_y})_0 & (\frac{\partial Z}{\partial r_z})_0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dS \\ dr_x \\ dr_y \\ dr_z \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} = \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad (3.173)$$

Where;

$$\frac{\partial X}{\partial S} = X + r_x Y - r_y Z \quad (3.174)$$

$$\frac{\partial Y}{\partial S} = -r_z X + Y + r_x Z \quad (3.175)$$

$$\frac{\partial Z}{\partial S} = r_y X - r_x Y + Z \quad (3.176)$$

$$\frac{\partial X}{\partial r_x} = 0 \quad (3.177)$$

$$\frac{\partial X}{\partial r_y} = -SZ \quad (3.178)$$

$$\frac{\partial X}{\partial r_z} = SY \quad (3.179)$$

$$\frac{\partial Y}{\partial r_x} = SZ \quad (3.180)$$

$$\frac{\partial Y}{\partial r_y} = 0 \quad (3.181)$$

$$\frac{\partial Y}{\partial r_z} = -SX \quad (3.182)$$

$$\frac{\partial Z}{\partial r_x} = -SY \quad (3.183)$$

$$\frac{\partial Z}{\partial r_y} = SX \quad (3.184)$$

$$\frac{\partial Z}{\partial z} = 0 \quad (3.185)$$

The matrix solution is:

$$X = (A^T A)^{-1} A^T L \quad (3.186)$$

$$X = \begin{bmatrix} dS \\ dr_x \\ dr_y \\ dr_z \\ dT_x \\ dT_y \\ dT_z \end{bmatrix} \quad (3.187)$$

The 3D Linearized Helmert transformation are calculated after first iteration, and assigned as the initial value of the second iteration, where;

$$S = S_0 + dS \tag{3.188}$$
$$r_x = r_{x0} + dr_x \tag{3.189}$$
$$r_y = r_{y0} + dr_y \tag{3.190}$$
$$r_z = r_{z0} + dr_z \tag{3.191}$$
$$T_x = T_{x0} + dT_x \tag{3.192}$$
$$T_y = T_{y0} + dT_y \tag{3.193}$$
$$T_z = T_{z0} + dT_z \tag{3.194}$$

3.3.7 Molodensky Transformation

The Molodensky method is a complex formula for the shift in latitude, longitude and height. The Molodensky Transformation has seven parameter transformations that include the three translation parameters, three rotation parameters and a scale parameter. [19]

Parameters are: S, ω, ∅, k, T_x, T_y and T_z .

It converts directly between two geographic coordinate systems without converting to an X, Y, Z system. This method has three translations (△ λ, △φ, △h) and the differences between the semi-major axes (△a) and the flattening (△f) of the two spheroids. The system automatically calculates the spheroid differences according to the datums, solve for △ λ and △φ. The amounts are added automatically by the system. [19]

The Molodensky transformation is based of all differences due to:

1- A shift in origin by a vector with components (dX, dY, dZ).
2- A difference in ellipsoids of size, (△a) and flattening (△f)

The Molodensky transformation gives directly the shifts in latitude, longitude and height. The angular values are in arc-seconds.

Starting from the general relationship between the geocentric coordinates (X, Y, Z) and to a specific ellipsoid with half-axes (a & b) geographic coordinates (φ, λ, h). [19]

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} (N+h).\cos\varphi\,.\cos\lambda \\ (N+h).\cos\varphi\,.\sin\lambda \\ \left(\frac{b^2}{a^2}.N+h\right).\sin\varphi \end{bmatrix} \tag{3.195}$$

The matrix obtained by differentiation of (3.195)

$$\begin{bmatrix} dX \\ dY \\ dZ \end{bmatrix} = \begin{bmatrix} \frac{\partial X}{\partial \varphi} & \frac{\partial X}{\partial \lambda} & \frac{\partial X}{\partial h} \\ \frac{\partial Y}{\partial \varphi} & \frac{\partial Y}{\partial \lambda} & \frac{\partial Y}{\partial h} \\ \frac{\partial Z}{\partial \varphi} & \frac{\partial Z}{\partial \lambda} & \frac{\partial Z}{\partial h} \end{bmatrix} . \begin{bmatrix} d\varphi \\ d\lambda \\ dh \end{bmatrix}$$

$$\begin{bmatrix} -(M+h).\sin\varphi\cos\lambda & -(N+h).\cos\varphi\cos\lambda & \cos\varphi\cos\lambda \\ -(M+h).\sin\varphi\cos\lambda & (N+h).\cos\varphi\cos\lambda & \cos\varphi\sin\lambda \\ (M+h).\cos\varphi & 0 & \sin\varphi \end{bmatrix} . \begin{bmatrix} d\varphi \\ d\lambda \\ dh \end{bmatrix} \quad (3.196)$$

$$= \begin{bmatrix} A(dX\_dGeo) \end{bmatrix} . \begin{bmatrix} d\varphi \\ d\lambda \\ dh \end{bmatrix}$$

For two differentially adjacent points $(X,Y,Z)_1$ and $(X,Y,Z)_2$ therefore, with (3.196) can be rewritten as:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_2 - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_1 = \begin{bmatrix} A(dX\_dGeo) \end{bmatrix}_1 . (\begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_2 - \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_1) \quad (3.197)$$

The index 1 indicates that calculated under linearization at the point $(\varphi, \lambda, h)_1$ matrix [A (dX_dGeo)]. The inversion of (3.197) provides:

$$\begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_2 - \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_1 = \begin{bmatrix} A(dX_{dGeo}) \end{bmatrix}_1^{-1} . (\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_2 - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_1) \quad (3.198)$$

Furthermore it is assumed that the position differences of $(X,Y,Z)_1$ and $(X,Y,Z)_2$ by a datum transformation with the datum parameter $d = (\varepsilon_x, \varepsilon_y, \varepsilon_z, s, t_x, t_y, t_z)$ for 3 rotations $d_{rotation} = (\varepsilon_x, \varepsilon_y, \varepsilon_z)$ a scale $d_s = s$ and three translations d = (t_x, t_y, t_z) will be achieved. The mater relation is obtained using classical relations for a nonlinear seven-parameter similarity transformation, so initially: [19]

$$\begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_2 = \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_1 + \begin{bmatrix} A(dX_{dGeo}) \end{bmatrix}_1^{-1} . (s.R(\varepsilon_x, \varepsilon_y, \varepsilon_z). \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_1 + t - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_2) \quad (3.199)$$

It is assumed by (3.196), that this Datum parameters are a non-linear, that (3.199) can be linearized at the point $d_0 = (\varepsilon_x = 0, \varepsilon_y = 0, \varepsilon_z = 0, s = 1, t_x = 0, t_y = 0, t_z = 0)$. Thus the obtained matrix (3.200): [19]

$$\begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_2 = \begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_1 - \left[A(dX_{dGeo})\right]_1^{-1} . \begin{bmatrix}X\\ Y\\ Z\end{bmatrix}_1 +$$

$$\left[A(dX_{dGeo})\right]_1^{-1} (\begin{bmatrix}X\\ Y\\ Z\end{bmatrix}_1 + \begin{bmatrix}0 & -Z_1 & Y_1 & |X_1| & 1 & 0 & 0\\ Z_1 & 0 & -X_1 & |Y_1| & 0 & 1 & 0\\ -Y_1 & X_1 & 0 & |Z_1| & 0 & 0 & 1\end{bmatrix} . \begin{bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_z\\ \Delta s\\ t_x\\ t_y\\ t_z\end{bmatrix} \quad (3.200)$$

An alternative for (3.200) is given in equation (3.201):

$$\begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_2 = \begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_1 - \left[A(dX_{dGeo})\right]_1^{-1} .(\begin{bmatrix}0 & -z_1 & y_1 & |X_1| & 1 & 0 & 0\\ z_1 & 0 & -x_1 & |Y_1| & 0 & 1 & 0\\ -y_1 & x_1 & 0 & |Z_1| & 0 & 0 & 1\end{bmatrix} . \begin{bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_z\\ \Delta s\\ t_x\\ t_y\\ t_z\end{bmatrix}$$

(3.201)

In (3.201) the $(X, Y, Z)_1$ inserted for in turn of (3.195) between the Cartesian coordinates $(X, Y, Z)_1$ and the geographical coordinates $(\varphi, \lambda, h)_1$, and the relation is obtained by multiplying them both ,so that final relation:

$$\begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_2 = \begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_1 + [Molodensky]_{(\varphi,\lambda,h)_1} . \begin{bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_z\\ \Delta s\\ t_x\\ t_y\\ t_z\end{bmatrix} \quad (3.202)$$

The position $(\varphi, \lambda, h)_1$ not only referenced to a different datum but at the same time there is another applicable reference ellipsoid, so to $(\varphi, \lambda, h)_1$ and $(\varphi, \lambda, h)_2$ belonging reference ellipsoid (a_1, b_1) and (a_2, b_2) differently dimensioned, so are on $(\varphi, \lambda, h)_2$ in addition to corrections $(\Delta\varphi, \Delta\lambda, \Delta h)_{(a,b)_1(a,b)_2}$ to add an ellipsoid dimensions transformations. We obtain thus in final general form: [19]

$$(\begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_2 + \begin{bmatrix}\Delta\varphi_{(a,b)_1(a,b)_2}\\ \Delta\lambda_{(a,b)_1(a,b)_2}\\ \Delta h_{(a,b)_1(a,b)_2}\end{bmatrix} - \begin{bmatrix}\varphi\\ \lambda\\ h\end{bmatrix}_1 + \begin{bmatrix}v_\varphi\\ v_\lambda\\ v_h\end{bmatrix} [Molodensky]_{(\varphi,\lambda,h)_1,i} . \begin{bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_z\\ \Delta s\\ t_x\\ t_y\\ t_z\end{bmatrix} \quad (3.203)$$

Where;

$$\Delta\varphi_{(a,b)_1(a,b)_2} = \varphi_{(a_1,b_1|(X,Y,Z)_1)} - \varphi_{(a_2,b_2|(X,Y,Z)_2)} \quad (3.204)$$

$$\Delta\lambda_{(a,b)_1(a,b)_2} = 0 \quad (3.205)$$

$$\Delta h_{(a,b)_1(a,b)_2} = h_{(a_1,b_1|(X,Y,Z)_1)} - h_{(a_2,b_2|(X,Y,Z)_2)} \quad (3.206)$$

$$[Molodensky]_{(\varphi,\lambda,h)_1} =$$

$$\begin{bmatrix} -\sin\lambda.\frac{a.W+h}{M+h} & \cos\lambda.\frac{a.W+h}{M+h} & 0 & \frac{-\sin\varphi.\cos\varphi.N.e^2}{M+h} & \frac{-\sin\varphi.\cos\lambda}{M+h} & \frac{-\sin\varphi.\sin\lambda}{M+h} & \frac{\cos\varphi}{M+h} \\ \frac{\sin\varphi.\cos\lambda.(N.(1-e^2)+h}{(N+h).\cos\varphi} & \frac{\sin\varphi.\sin\lambda.(N.(1-e^2)+h}{(N+h).\cos\varphi} & -1 & 0 & \frac{-\sin\lambda}{(N+h).\cos\varphi} & \frac{\cos\lambda}{(N+h).\cos\varphi} & 0 \\ -N.e^2.\sin\varphi.\cos\varphi.\sin\lambda & N.e^2.\sin\varphi.\cos\varphi.\cos\lambda & 0 & h+a.W & \cos\varphi.\cos\lambda & \cos\varphi.\sin\lambda & \sin\varphi \end{bmatrix}$$

(3.207)

Where;

$$W = \frac{a}{N} = \sqrt{1 - e^2.sin^2\varphi} \quad (3.208)$$

$$e^2 = \frac{a^2-b^2}{a^2} \quad (3.207)$$

Here;

h: ellipsoid height (meters)
φ: latitude in the source system
λ: longitude in the source system
a: semi-major axis of the spheroid (meters)
b: semi-minor axis of the spheroid (meters)
f: flattening of the spheroid
e: eccentricity of the spheroid
M & N: the meridian and prime vertical radius of curvature, respectively, at a given latitude.

The advantage of Molodensky over similarity transformation is that we can easily separate horizontal position (λ,φ) and the vertical position (h), while there is no need to refer to the Geocentric coordinates (X.Y.Z), in addition on difference in the ellipsoid used are included in the transformation.

3.4. Precision & Blunder detection

3.4.1. Precision of indirectly determined quantities

Consider an adjustment involving weighted observation equation, the matrix form for the system of weighted observation equations is:

$$W.A.X = W.L + W.V \quad (3.208)$$

And the least squares solution of the weighted observation equations is given by:

$$X = (A^T.W.A)^{-1}.A^T.W.L \quad (3.209)$$

In this equation, X contains the most probable values for the unknowns, whereas the true values are X_{true}. The true value differ from X by some small amount ΔX, such that:

$$X + \Delta X = X_{true} \quad (3.210)$$

Where ΔX presents the true errors in the adjusted values.

Consider now a small incremental change, ΔL, in the observed values, L, which changes X to its true value, X + ΔX. Then equation (3.209) becomes:

$$X + \Delta X = (A^T.W.A)^{-1}.A^T.W.(L + \Delta L) \quad (3.211)$$

Expanding equation (3.211) yields:

$$X + \Delta X = (A^T.W.A)^{-1}.A^T.W.L + (A^T.W.A)^{-1}.A^T.W.\Delta L \quad (3.212)$$

Note in equation (3.210) that X = $(A^T.W.A)^{-1}.A^T.W.L$, and thus subtracting this from equation (3.212) yields:

$$\Delta X = (A^T.W.A)^{-1}.A^T.W.\Delta L \quad (3.213)$$

Recognizing ΔL as the errors in the observation, equation (3.213) can be rewritten as:

$$\Delta X = (A^T.W.A)^{-1}.A^T.W.V \quad (3.214)$$

Where the vector of residuals V is substituted for ΔL. Now let:

$$B = (A^T.W.A)^{-1}.A^T.W \quad (3.215)$$

Then

$$\Delta X = B.V \quad (3.216)$$

Multiplying both sides of equation (3.216) by their transpose results in:

$$\Delta X.\Delta X^T = (B.V)(B.V)^T \quad (3.217)$$

Applying the matrix property $(BV)^T = V^T.B^T$ to equation (3.217) yields:

$$\Delta X.\Delta X^T = B.V.V^T.B^T \tag{3.218}$$

The expanded left side of equation (3.218) is:

$$\Delta X.\Delta X^T = \begin{bmatrix} \Delta x_1^2 & \Delta x_1\Delta x_2 & \Delta x_1\Delta x_3 & \cdots & \Delta x_1\Delta x_n \\ \Delta x_2\Delta x_1 & \Delta x_2^2 & \Delta x_2\Delta x_3 & \cdots & \Delta x_2\Delta x_n \\ \Delta x_3\Delta x_1 & \Delta x_3\Delta x_2 & \Delta x_3^2 & \cdots & \Delta x_3\Delta x_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \Delta x_n\Delta x_1 & \Delta x_n\Delta x_2 & \Delta x_n\Delta x_3 & \cdots & \Delta x_n^2 \end{bmatrix} \tag{3.219}$$

Also, the expanded right side of equation (3.218) is:

$$\Delta X.\Delta X^T = \begin{bmatrix} v_1^2 & v_1v_2 & v_1v_3 & \cdots & v_1v_m \\ v_2v_1 & v_2^2 & v_2v_3 & \cdots & v_2v_m \\ v_3v_1 & v_3v_2 & v_3^2 & \cdots & v_3v_m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_mv_1 & v_mv_2 & v_mv_3 & \cdots & v_m^2 \end{bmatrix} \tag{3.220}$$

Assume that it is possible to repeat the entire sequence of observation many times, say a times, and that each time a slightly different solution occurs, yielding a different set of X's. Averaging these sets, the left side of equation (3.218) becomes: [2]

$$\frac{1}{a}\sum(\Delta X)(\Delta X^T) = \begin{bmatrix} \frac{\sum \Delta x_1^2}{a} & \frac{\sum \Delta x_1\Delta x_2}{a} & \cdots & \frac{\sum \Delta x_1\Delta x_n}{a} \\ \frac{\sum \Delta x_2\Delta x_1}{a} & \frac{\sum \Delta x_2^2}{a} & \cdots & \frac{\sum \Delta x_2\Delta x_n}{a} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sum \Delta x_n\Delta x_1}{a} & \frac{\sum \Delta x_n\Delta x_2}{a} & \cdots & \frac{\sum \Delta x_n^2}{a} \end{bmatrix} \tag{3.221}$$

If a is large, the terms in equation (3.221) are the variance and covariance, and it can be rewritten as:

$$\begin{bmatrix} S_{x_1}^2 & S_{x_1x_2} & \cdots & S_{x_1x_n} \\ S_{x_2x_1} & S_{x_2}^2 & \cdots & S_{x_2x_n} \\ \vdots & \vdots & \ddots & \vdots \\ S_{x_nx_1} & S_{x_nx_2} & \cdots & S_{x_n}^2 \end{bmatrix} = S_{xx}^2 \tag{3.222}$$

Also, considering a sets of observation, equation (3.220) becomes:

$$B\begin{bmatrix} \frac{\sum v_1^2}{a} & \frac{\sum v_1 v_2}{a} & \cdots & \frac{\sum v_1 v_m}{a} \\ \frac{\sum v_2 v_1}{a} & \frac{\sum v_2^2}{a} & \cdots & \frac{\sum v_2 v_m}{a} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sum v_m v_1}{a} & \frac{\sum v_m v_2}{a} & \cdots & \frac{\sum v_m^2}{a} \end{bmatrix} B^T \quad (3.223)$$

Recognizing the diagonal terms as variances of the quantities observed, $S_{l_i}^2$, off-diagonal terms as the covariances, $S_{l_i l_j}$, and the fact that the matrix is symmetric, equation (3.223) can be rewritten as: [2]

$$B\begin{bmatrix} S_{l_1}^2 & S_{l_1 l_2} & \cdots & S_{l_1;m} \\ S_{l_2 l_1} & S_{l_2}^2 & \cdots & S_{l_2 l_m} \\ \vdots & \vdots & \ddots & \vdots \\ S_{l_m l_1} & S_{l_m l_2} & \cdots & S_{l_m}^2 \end{bmatrix} \quad (3.224)$$

The weight of an observation is inversely proportional to its variance. Also, the variance of an observation of weight w can be expressed in terms of the reference variance as:

$$S_i^2 = \frac{S_0^2}{W_i} \quad (3.225)$$

Recall that $W = Q^{-1} = \sigma_0^2 \Sigma^{-1}$. Therefor, $\Sigma = \sigma_0^2 W^{-1}$, and by substituting equation (3.225) into matrix (3.224) and replacing σ_0 with S_0 yields:

$$S_0^2 . B . W_{ll}^{-1} . B^T \quad (3.226)$$

Substituting equation (3.225) into equation (3.226) gives:

$$S_0^2 . B . W^{-1} . B^T = S_0^2 (A^T . W . A)^{-1} . A^T . W . W^{-1} W^T \; [(A^T . W . A)^{-1}]^T \quad (3.227)$$

Since the normal and weight matrices are symmetric, it follows that:

$$[(A^T . W . A)^{-1}]^T = (A^T . W . A)^{-1} \quad (3.228)$$

Also, since the weight matrix W is symmetric, $W^T = W$, and thus equation (3.227) reduce to:

$$S_0^2 (A^T . W . A)^{-1} . (A^T . W . A) . (A^T . W . A)^{-1} = S_0^2 (A^T . W . A)^{-1} \quad (3.229)$$

Equation (3.222) is the left side of equation (3.218), for which equation (3.229) is the right. That is:

$$S_x^2 = S_0^2 (A^T . W . A)^{-1} = S_0^2 N^{-1} = S_0^2 Q_{xx} \quad (3.230)$$

In least square adjustment, the matrix S_x^2 of equation (3.230) is known as the variance- covariance matrix, or simply the covariance matrix, and Q_{xx} is called the cofactor matrix for the adjusted unknowns. When multiplied by S_0^2, diagonal elements of the cofactor matrix yield variances of the adjusted quantities, and the off-diagonal elements multiplied by S_0^2 yields covariances. From equation (3.230), the estimated standard deviation S_i for any unknown parameter having been computed from a system of observation equation is expressed as: [2]

$$S_i = S_0\sqrt{q_{x_ix_i}} \tag{3.231}$$

Where $q_{x_ix_i}$ is the diagonal element (from the i^{th} row and i^{th} column) of the Q_{xx} matrix, which as noted in equation (3.230) is equal to the inverse of the matrix of normal equations. Since the normal equation matrix is symmetric matrix (i.e., element ij = element ji). [2]

3.4.2. Blunder detection

3.4.2.1. Development of the covariance matrix for the residuals

The concept of statistical blunder in surveying was introduced in the mid-1960s and utilizes the cofactor matrix for the residuals. To develop this matrix, the adjustment of a linear problem can be expressed in matrix form as: [2]

$$L + V = AX + C \tag{3.232}$$

Where C is a constant vector, A the coefficient matrix, X the estimated parameter matrix, L the observation matrix, and V the residual vector. Equation (3.232) can be rewritten in terms of V as:

$$V = AX - T \tag{3.233}$$

Where $T = L - C$, which has a covariance matrix of $W^{-1} = S^2Q_{ll}$. The solution of equation (3.233) results in the expression:

$$X = (A^T.W.A)^{-1}.(A^T.W.T) \tag{3.234}$$

Letting ε represent a vector of true errors for the observation, equation (3.232) can be written as:

$$L + \varepsilon = A\bar{X} + C \tag{3.235}$$

Where $\bar{X}$ is the true value for the unknown parameter X and thus:

$$\mathrm{T} = L - C = A\bar{X} + \varepsilon \tag{3.236}$$

Substituting equation (3.234) and (3.236) into equation (3.233) yields:

$$V = A.(A^T.W.A)^{-1}.A^T.W.(A.X + \varepsilon) - (A.\bar{X} + \varepsilon) \tag{3.237}$$

Expanding equation (3.237) result is:

$$V = A.(A^T.W.A)^{-1}.A^T.W.\varepsilon - \varepsilon + A.(A^T.W.A)^{-1}.A^T.W.A.\bar{X}A.\bar{X})$$

(3.238)

Since $A.(A^T.W.A)^{-1} = A^{-1}.W^{-1}.A^{-T}$, equation (3.238) can be simplified to:

$$V = A.(A^T.W.A)^{-1}.A^T.W.\varepsilon - \varepsilon \quad (3.239)$$

Factoring $W\ \varepsilon$ from equation (3.239) yields

$$V = -(W^{-1} - A.(A^T.W.A)^{-1}.A^T).W.\varepsilon \quad (3.240)$$

Recognizing $(A^T.W.A)^{-1} = Q_{xx}$ and defining $Q_{vv} = W^{-1} - AQ_{xx}A^T$, equation (3.240) can be rewritten as:

$$V = -Q_{vv}.W.\varepsilon \quad (3.241)$$

Where $Q_{vv} = W^{-1} - AQ_{xx}A^T = W^{-1} - Q_{ll}$.

If we let R be the product of W and Q_{vv}, the R matrix is both singular and idempotent. Being singular, it has no inverse. When a matrix is idempotent, the following properties exist for the matrix; (a) the square of the matrix is equal to the original matrix (i.e., $R.R = R$); (b) every diagonal element is between zero and 1; (c) the sum of the diagonal elements, known as the trace of the matrix, equals the degrees of freedom in the adjustment, the latter property expressed mathematically as: [2]

$$r_{11} + r_{22} + \cdots + r_{mm} = \text{degrees of freedom} \quad (3.242)$$

(d) the sum of the square of the elements in any single row or column equals the diagonal element, that is:

$$r_{ii} = r_{i1}^2 + r_{i2}^2 + \cdots + r_{im}^2 = r_{1i}^2 + r_{2i}^2 + \cdots + r_{mi}^2 \quad (3.243)$$

Now consider the case when all observations have no error except for a particular observation l_i that contains a blunder of size Δl_i. A vector of the true errors is expressed as:

$$\Delta\varepsilon = \Delta l_i \varepsilon_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \Delta l_i \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \Delta l_i \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.244)$$

If the original observations are uncorrelated, the specific correction for Δv_i can be expressed as:

$$\Delta v_i = -q_{ii} w_{ii} \Delta l_i = -r\Delta l_i \quad (3.245)$$

Where q_{ii} is the i^{th} diagonal of the Q_{vv} matrix, w_{ii} the i^{th} diagonal term of the weight matrix W, and $r_i = q_{ii}w_{ii}$ the observation's redundancy number.

When the system has a unique solution, r_i will equal zero, and if the observation is overconstrained, r_i will equal 1. The redundancy numbers provide insight into the geometric strength of the adjustment. An adjustment that in general has low redundancy numbers will have observation that lack sufficient checks to isolate blunders, and thus the chance for undetected blunders to exist in the observations is high. Conversely, a high overall redundancy number enables a high level of internal checking of the observations, and thus there is a lower chance of accepting observations that contains blunders. The quotient or r/m is called the relative redundancy of the adjustment, where r is the total number of redundant observations on the system and m is the number of observations. [2]

3.4.2.2. Data Snooping

Equation (3.241) defines the covariance matrix for the vector of residuals, v_i. From this the standardized residual is computed using the appropriate diagonal element of the Q_{vv} matrix as: [2]

$$\bar{v}_i = \frac{v_i}{\sqrt{q_{ii}}} \tag{3.246}$$

Where $\bar{v}_i$ is the standardized residual, v_i the computed residual, and q_{ii} the diagonal element of the Q_{vv} matrix. Using the Q_{vv} matrix, the standard deviation in the residual is $S_0\sqrt{q_{ii}}$. Thus if the denominator of equation (3.246) is multiplied by S_0, a t statistic is defined. If the residual differs significantly from zero, the observation used to derive the statistics is considered to be blunder. The test statistic for this hypothesis test is: [2]

$$t_i = \frac{v_i}{S_0\sqrt{q_{ii}}} = \frac{v_i}{S_v} = \frac{\bar{v}_i}{S_0} \tag{3.247}$$

When a blunder is present in the data set, the t distribution is shifted, and a statistical test for this shift may be performed. As with any statistical test, two types of errors can occur. A type **I** error occurs when data are rejected that do not contain blunders, and a type **II** error occurs when a blunder is not detected in a data set where one is actually present. The rejection criteria are represented by the vertical line in figure (3.9), and their corresponding significance levels are shown in table (3.1).

Thus, the approach is to use a rejection level given by a t distribution with $r - 1$ degrees of freedom. The observation with the largest absolute value of t_i as given by equation (3.248) is rejected when it is greater than the rejection level. [2]

The observation is rejected when:

$$\frac{|v_i|}{S_0\sqrt{q_{ii}}} > rejection\ level \tag{3.248}$$

Table (3.1): Rejection Criteria with Corresponding Significance Levels [2]

α	$1-\alpha$	β	$1-\beta$	Rejection Criteria
0.005	0.95	0.80	0.20	2.8
0.001	0.999	0.80	0.20	4.1
0.001	0.999	0.999	0.001	6.6

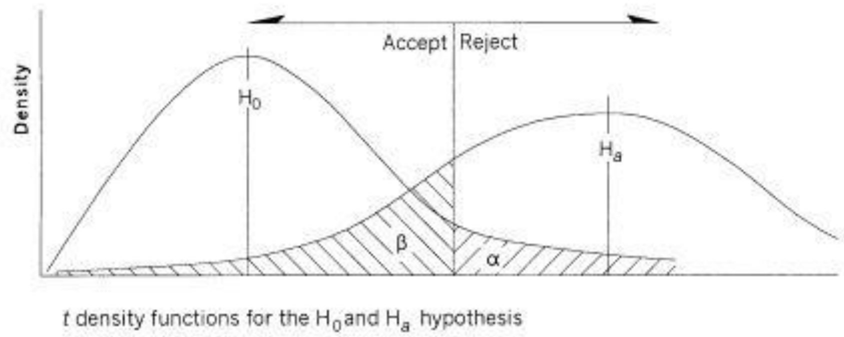

Figure (3.9): Effects of a blunder on the t distribution [2]

Since the existence of any blunder in a set of data will affect the remaining observations, and since equation (3.249) is dependent on S_0, whose value was computed from data containing blunders, all observations that are detected as blunders should not be removed in a single pass. Instead, only the largest or largest blunders from independent groups of observations should be deleted. Furthermore, since equation (3.249) is dependent on S_0, it is possible to rewrite the equation so that it can be computed during the final iteration of a nonlinear adjustment. In this case the appropriate equations is: [2]

$$\bar{v}_i = \frac{|v_i|}{\sqrt{q_{ii}}} > S_0 * rejection\ level \tag{3.249}$$

A summary of procedures for this manner of blunder detection is as follows: [2]

Step 1: Located all standardized residual that meet the rejection criteria of equation (3.248) or (3.249).

Step 2: Remove the largest detected blunder or unrelated blunder groups.

Step 3: Rerun the adjustment.

Step 4: Continue steps 1 through 3 until all detected blunders are removed.

Step 5: If more than one observation is removed in steps 1 through 4, reenter the observation in the adjustment in a one-at-a-time fashion. Check the observation after each adjustment to see if it is again detected as a blunder. If it is, remove it from the adjustment or have that observation reobserved.

Again it should be noted that this form of blunder detection is sensitive to improper relative weighting in observations. Thus, it is important to use weights that are reflective of the observational errors. [2]

CHAPTER 4

THE CONVERTER

4.1. Introduction

4.2. Decimal to Radian conversion and vice versa.

4.3. Decimal to Degrees conversion and vice versa.

4.4. Degrees to Radian conversion and vice versa.

4.1. Introduction

The Geo Transform V 2.0 has a conversion tool, which is a tool that helps the user to convert from a certain angular format to another and vice versa.
Currently, the supported angular formats are:

1- Decimal Degree (DD).
2- Degrees/Minutes/Seconds (DMS).
3- Radian.

Decimal degree, degrees/minutes/seconds and radian are units for measuring angles. DMS and radian are the ones are most likely to encounter in common projects. An angle between two non-parallel lines can be measured numerically in either a decimal format or degree format. Degrees are used to express directionality and angle size. But since the degrees are not actually numbers and can't be used in mathematical computations. Radian is used, because it's simply the amount we turn around a circle, calculated as the length of the arc, divided by the radius of the circle.

4.2. Decimal to Radian conversion and vice versa.

The mathematical method to convert Decimal to Radian is shown in the following formula:

$$DD * \pi / 180 = \text{radian angle} \qquad (4.1)$$

The reverse formula is:

$$R * 180 / \pi = \text{decimal degree angle} \qquad (4.2)$$

Where;

DD: the angle in decimal degrees format
R: the angle in Radian format

$$\pi = 3.141592654 \qquad (4.3)$$

Example (4.1):

Convert 30.263889° angle to radian:

$$R = 30.263889^\circ * \pi / 180 = 0.528204507577$$

Example (4.2):

Convert 0.528204507577 angle to decimal degree:

$$DD = 0.528204507577 * 180 / \pi = 30.263889^\circ$$

4.3. Decimal to Degrees conversion and vice versa.

The mathematical method to convert Decimal to Degrees is shown in the following formula:

$$d = \text{integer}(DD) \quad (4.4)$$
$$m = \text{integer}((DD - d) \times 60) \quad (4.5)$$
$$s = (DD - d - m/60) \times 3600 \quad (4.6)$$
$$DMS = d + m + s \quad (4.7)$$

The reverse formula is:

$$DD = d + m / 60 + s / 3600 \quad (4.8)$$

Where;

DD: the angle in decimal degree format.
d: the angel in integer degrees.
m: minutes.
s: seconds.
DMS: the angle in Degree-Minute-Second format.

Example (4.3):

Convert 30.263888889° angle to degrees-minutes-seconds (DMS):

$$d = \text{integer}(30.263888889^\circ) = 30^\circ$$
$$m = \text{integer}((DD - d) \times 60) = 15'$$
$$s = (DD - d - m/60) \times 3600 = 50''$$

So;

$$30.263888889^\circ = 30^\circ\ 15'\ 50''$$

Example (4.4):

Convert 30 degrees 15 minutes and 50 seconds angle to decimal degrees:

$$DD = 30^\circ + 15'/60 + 50''/3600 = 30.263888889^\circ$$

4.4. Degrees to Radian conversion and vice versa.

This method is a compound between the three formats (decimal, DMS and radian), it depends on converting the angle from DMS to decimal and then finally to radian as shown in equation (4.8) and (4.9). Also the conversion from radian to decimal then finally to DMS is show in equation (4.10) through (4.14):

$$DD = d + m / 60 + s / 3600 \quad (4.9)$$

$$DD * \pi / 180 = \text{radian angle} \quad (4.10)$$

The reverse formula is:

$$R * 180 / \pi = DD \quad (4.11)$$

$$d = \text{integer}(DD) \quad (4.12)$$

$$m = \text{integer}((DD - d) \times 60) \quad (4.13)$$

$$s = (DD - d - m/60) \times 3600 \quad (4.14)$$

$$DMS = d + m + s \quad (4.15)$$

Where;

DD: the angle in decimal degree format.
d: the angel in integer degrees.
m: minutes.
s: seconds.
DMS: the angle in Degree-Minute-Second format.
R: the angle in Radian format

$$\pi = 3.141592654 \quad (4.16)$$

Example (4.5):

Convert 30 degrees 15 minutes and 50 seconds angle to radian:

$$DD = 30° + 15'/60 + 50''/3600 = 30.263888889°$$

$$\text{Radian} = 30.263888889 * \pi / 180 = 0.528204507577$$

Example (4.6):

Convert 0.528204507577 in radian to degrees/minutes/seconds:

$$DD = 0.528204507577 * 180 / \pi = 30.263889°$$

$$d = \text{integer}(30.263888889°) = 30°$$

$$m = \text{integer}((DD - d) \times 60) = 15'$$

$$s = (DD - d - m/60) \times 3600 = 50''$$

So;

$$30.263888889° = 30°\ 15'\ 50''$$

CHAPTER 5

USING THE GEO-TRANSFORM V 2.0

5.1. Introduction

5.2. Using the Geo Transform for Coordinates Transformations.

5.3. Using the Geo Transform for Datum Transformations.

5.4. Using the Geo Transform for Angular Conversion.

USING THE GEO-TRANSFORM

5.1. Introduction

The geo transform is a tool programmed using C# programming language. This tool has the following functionalities; the first is to transform from Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z), where the same ellipsoid is used. The second is to transform Geographic coordinates (λ,φ) to 2D projected Grid coordinates (Easting, Northing), the used map projection in this transformation tool are: Cassini, Transverse Mercator (TM), Universal Transverse Mercator (UTM). The third is to apply the reverse transformation of the first and second functionalities.
In addition, a separate datum transformation tool is to be built to calculate the datum transformation parameters using different datum transformation methods similarity 7-parameters, similarity 6-parameters, similarity 3-parameters, Molodensky transformation and Helmert transformations.

In this tool three basic map projections are used; Cassini, Transverse Mercator(TM), Universal Transverse Mercator (UTM). In Palestine there are four map projection and coordinate systems have been used, these are; Palestine_1923_Grid, Palestine_1923_Belt, Palestine_1923_Israel_CS_Grid and Israel_TM_Grid. Where each coordinate system has its own projection parameters.

The Geo-Transform is a program capable of;

1. Transform from Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z), where the same ellipsoid is used.
2. Transform from Geocentric coordinates (X,Y,Z) to Geographic coordinates (λ,φ,h), where the same ellipsoid is used.
3. Transform from Geographic coordinates (λ,φ) to projected Grid coordinates (Easting, Northing), using Cassini, Transverse Mercator (TM), Universal Transverse Mercator (UTM).
4. Transform from projected Grid coordinates (Easting, Northing) to Geographic coordinates (λ,φ), using Cassini, Transverse Mercator (TM), Universal Transverse Mercator (UTM).
5. Calculate the datum transformation parameters using different transformation method; similarity 7-parameters, Molodensky transformation and Helmert transformations.
6. Transform the coordinates between different coordinates system.

5.2. Using the Geo Transform for Coordinates Transformations.

The use of the Geo-Transform program & its different steps are shown in the following sections:

5.2.1. Transform From Geographic to Geocentric Coordinates for a Single Point:

1. Open the Geo-Transform program.
2. Click on Coordinate Transformation button.

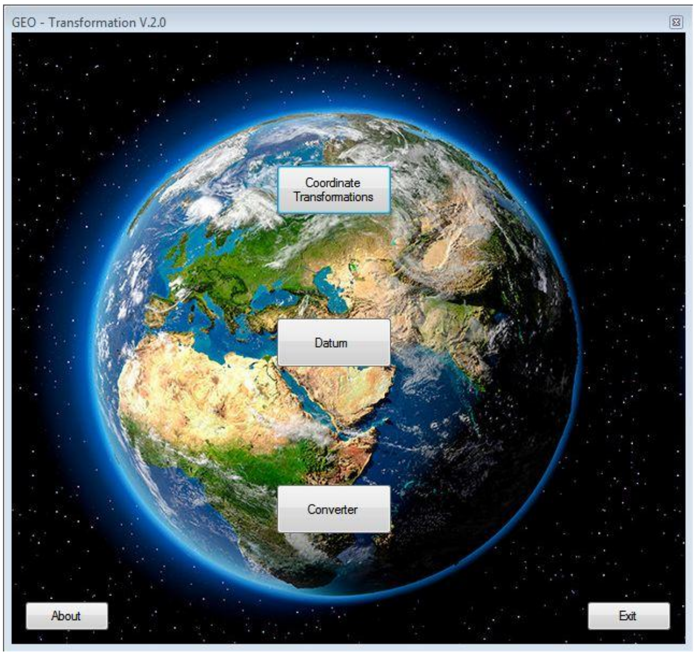

Figure (5.1): Geo-Transform Start Page

3. Choose a single point key.
4. Choose the lon, lat, h➔X,Y,Z key.
5. Enter the value of lon, lat and their direction , then enter the value of h. The direction of lon is East or West, but the direction of lat is North or south.
6. Select the ellipsoid you need, and then the value of a & b will appear according to the type of ellipsoid you select, but if you choose the user define you will enter the type of ellipsoid & the value of a & b.
7. Press the calculate key to have the result.

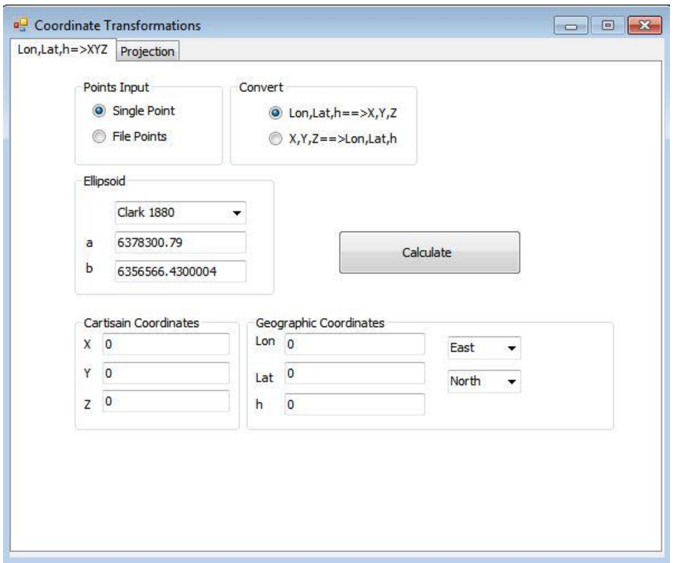

Figure (5.2): lon,lat,h➔X,Y,Z Dialog

Example (5.1):

1. In this example choose a single point.

2. Choose lon, lat, h ➔ X,Y,Z.

3. Enter the value $lon = 35$ & the direction of it east, then enter the value of $lat = 33$ & its direction north and enter the value of $h = 1000$.

4. Select Clarke 1880 ellipsoid, then the value of a & b will appear.

5. Press Calculate button, then the value of X,Y,Z will appear as fig (5.3).

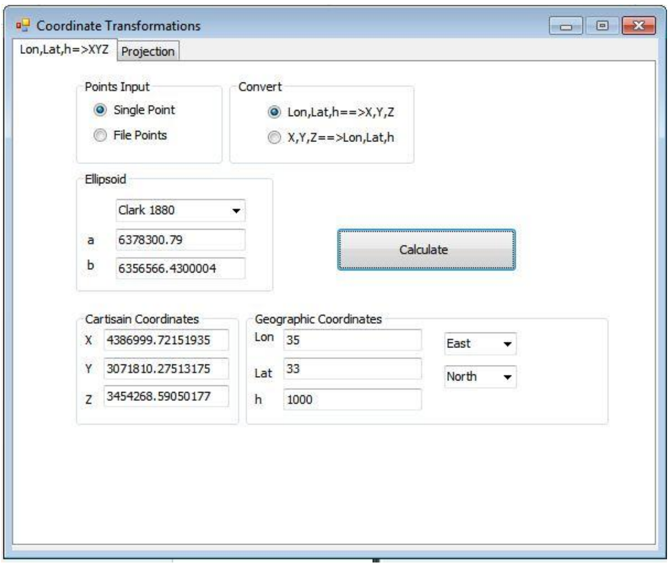

Figure (5.3): lon,lat,h➔X,Y,Z Dialog, Example (5.1)

5.2.2. Transform from Geocentric to Geographic Coordinates for a Single Point:

1. Choose a single point key.
2. Choose the X,Y,Z➔lon, lat, h.
3. Enter the value of the variables X Y Z.
4. Select the ellipsoid needed & the value of a & b will appear according to the type of the ellipsoid.
5. When finishing from entering the information, press calculate key & the value of lon, lat, h will appear with their direction as shown in figure (5.4).

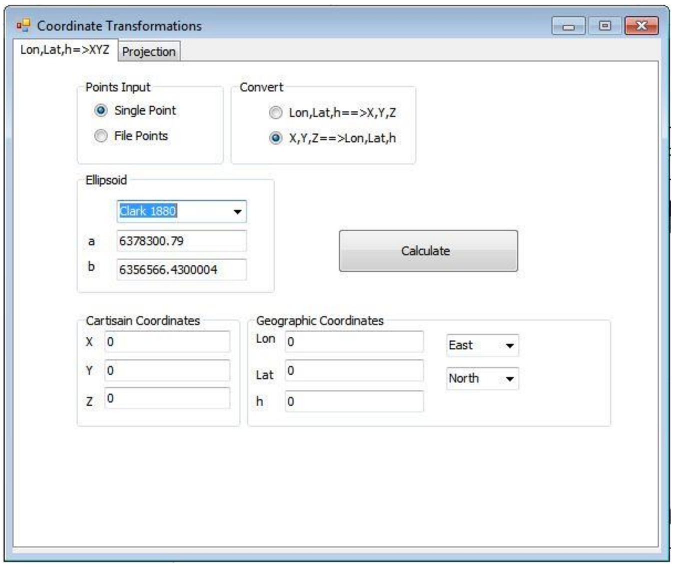

Figure (5.4): X,Y,Z➔ lon,lat,h Dialog

Example (5.2):

1. In this example choose X Y Z ➔ lon lat h.
2. Choose a single point.
3. Enter the value of X=3195088.60587309, Y=3195088.60587309, Z=4487800.65749324.
4. Select Clarke 1880 ellipsoid & the value of a & b will appear directly.
5. Press calculate key, then the result will appear as shown in figure (5.5).

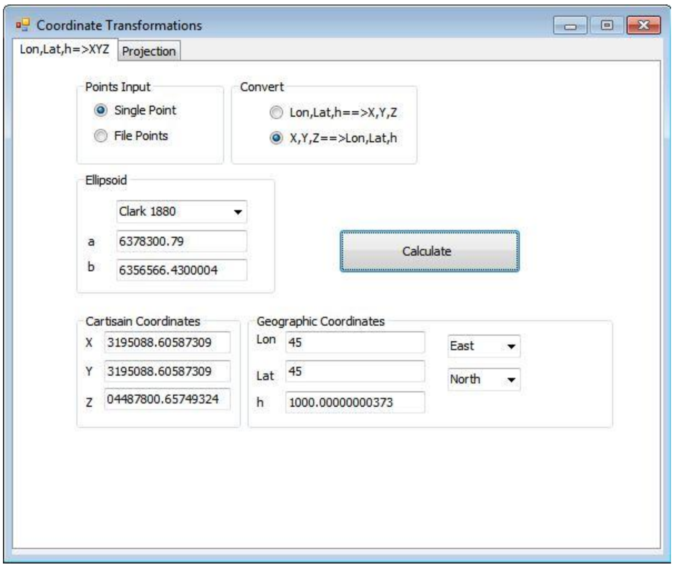

Figure (5.5): X,Y,Z➔lon,lat,h Dialog, Example (5.2)

5.2.3. Transform form Geographic to Projected Coordinates for a Single Point:

1. Choose the Single Point to enter one point.
2. Choose the Projection Direct key.
3. Enter the value of lat & its direction, & the value of lon & its direction.
4. Choose the coordinate system needed, if the choice is:
   - Palestine 1923 Grid the projection will be Cassini & the ellipsoid will be Clarke 1880.
   - Palestine -1923-Israel –CS-Grid, the projection will be Cassini & the ellipsoid will be Clarke 1880.
   - UTM the projection will be Universal Transverse Mercator & the ellipsoid will be enabling & you will enter zone number.
   - Palestine -1923-Belt, then the projection will be Transverse Mercator & the ellipsoid will be Clarke 1880.
   - Israel-TM-Grid, then the projection will be Transverse Mercator, & the ellipsoid will be GRS80.
   - Users define, you will choose Cassini projection or Transverse Mercator, & the ellipsoid will be enabling.
5. Select the ellipsoid needed & the value of a & b will appear directly
6. Enter the value of the projection parameter FE, FN, lat0, lon0, k0
7. Press the calculate key to have the result.

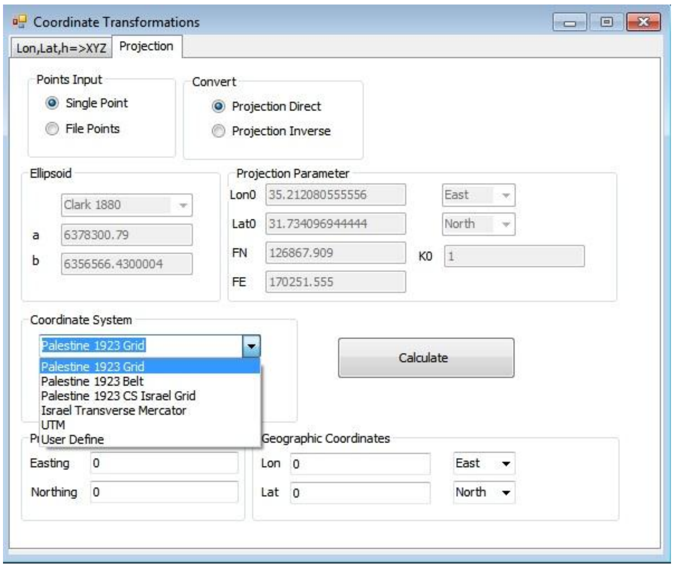

Figure (5.6): Direct Projection Dialog

Example (5.3): If the selection is Palestine-1923-Grid

1. Choose a single point.
2. Select Projection Direct key.
3. Enter the value of $lat = 33$ & its direction north, the value of $lon = 35$ & its direction east.
4. Select the Coordinate System Palestine -1923-Grid.
5. The ellipsoid will be Clarke 1880 directly.
6. Press calculate button to show the result, as shown in figure (5.7).

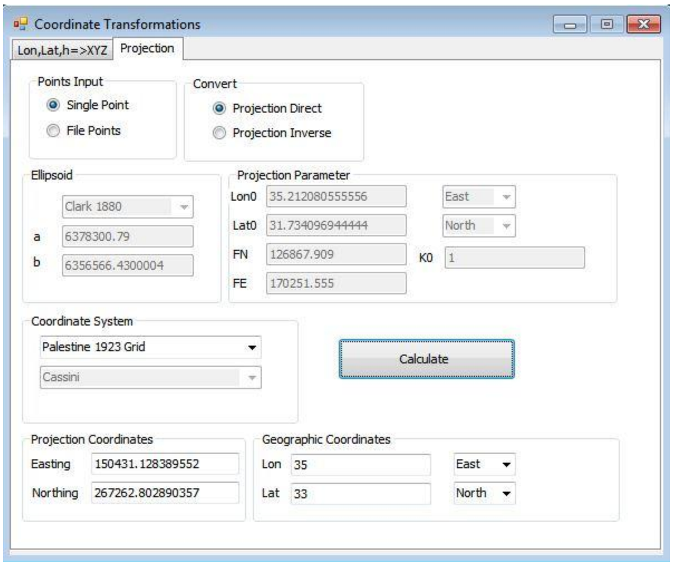

Figure (5.7): Direct Projection Dialog, Example (5.3)

5.2.4. Transform from Projected to Geographic Coordinates for a Single Point:

1. Choose the Single Point to enter one point.
2. Choose the Projection Inverse key.
3. Enter the value of lat & its direction, & the value of lon & its direction.
4. Choose the coordinate system needed.
5. Select the ellipsoid needed & the value of a & b will appear directly.
6. Enter the value of the projection parameter FE, FN, lat0, lon0, k0.
7. Press the calculate key to have the result.

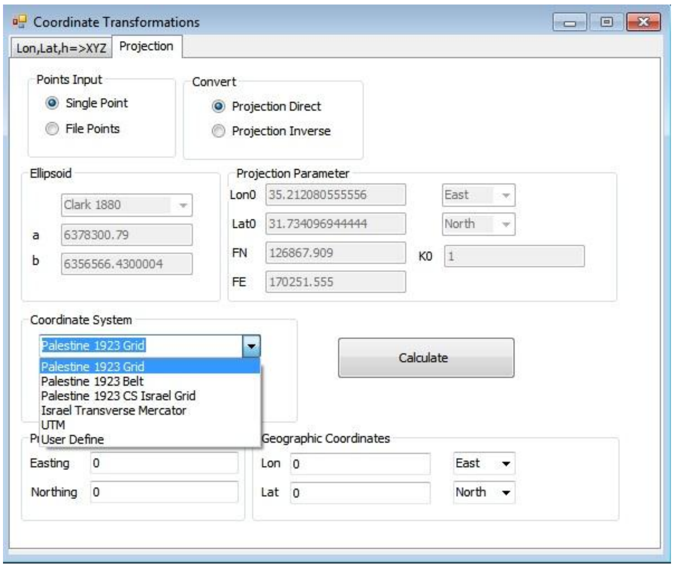

Figure (5.8): Inverse Projection Dialog

Example (5.4): if the selection is Israel Transverse Mercator

1. In Points Input check Single Point.
2. Select Projection Inverse key.
3. Enter the value of Easting=8962.766 & Northing=547567.301.
4. Select the coordinate system Israel Transverse Mercator.
5. The ellipsoid will be Transverse Mercator directly.
6. The value of projection parameters will be entered automatically too.
7. Press the calculation key to show the result, as shown in figure (5.9).

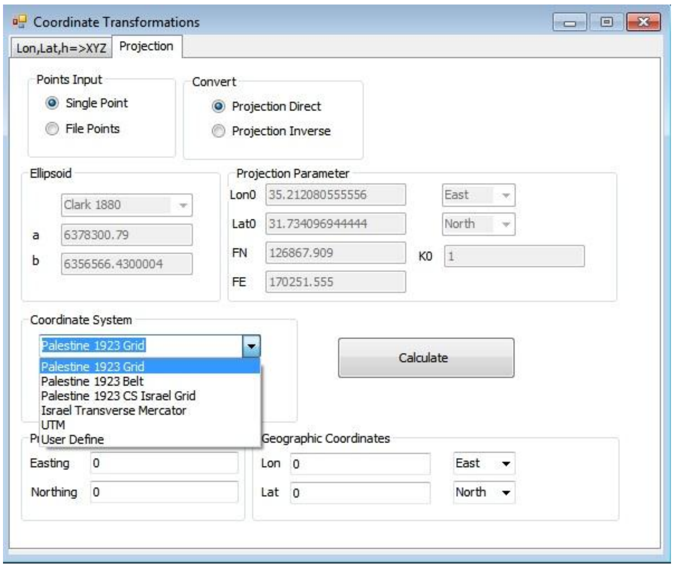

Figure (5.8): Inverse Projection Dialog, Example (5.4)

5.2.5. Transform from Geographic to Geocentric Coordinates for a Point File:

1. Select File Points.
2. Select lon, lat, h➔X, Y, Z key.
3. Select Input File button.

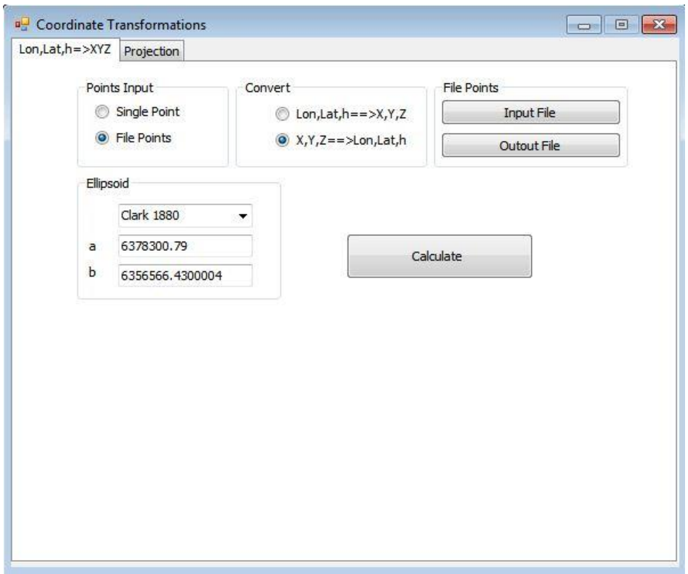

Figure (5.10): lon, lat, h ➔ X,Y,Z

4. Select InputLonLath file.

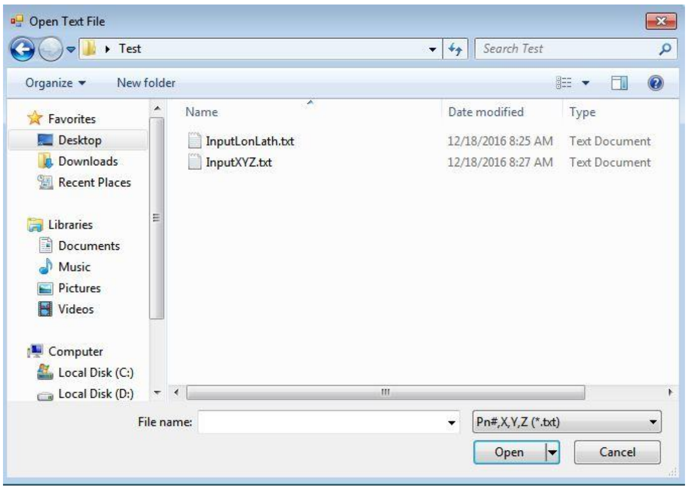

Figure (5.11): Open Dialog

5. Press open button.

6. Select the ellipsoid needed from the main dialog.

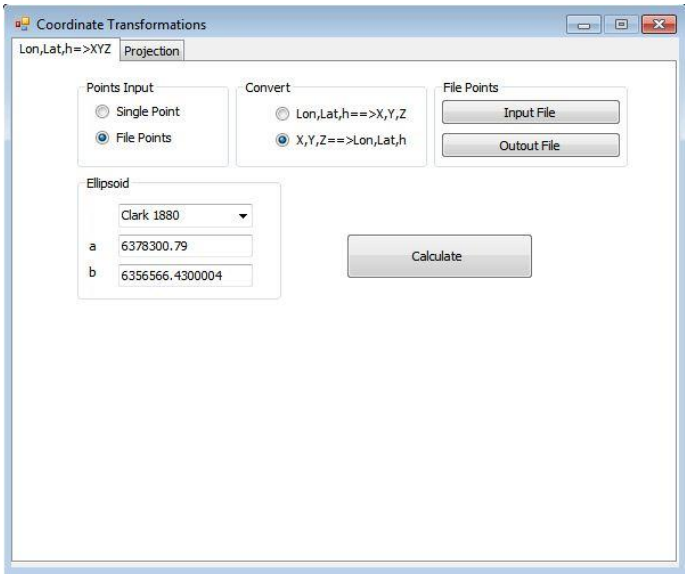

Figure (5.12): lon,lat,h➔X,Y,Z Dialog

7. Click on the Calculate button

8. Select Output File button to open Save As window.

9. This window will appear as in figure (5.13), write the file name OutputXYZ then click save.

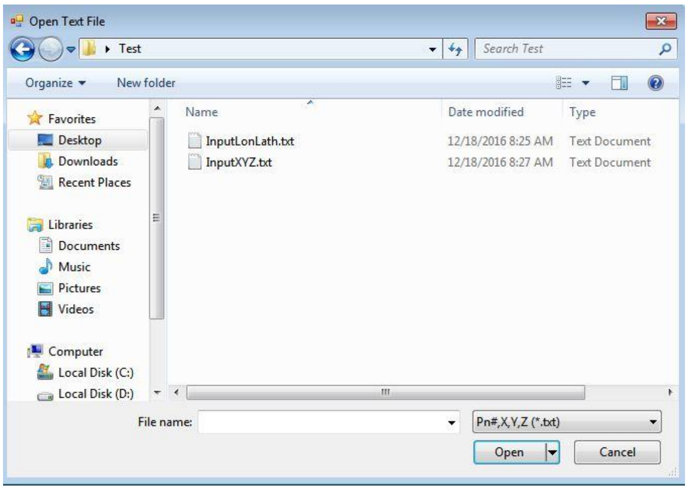

Figure (5.13): Save as Dialog

9. The result will appear in the file saved in.

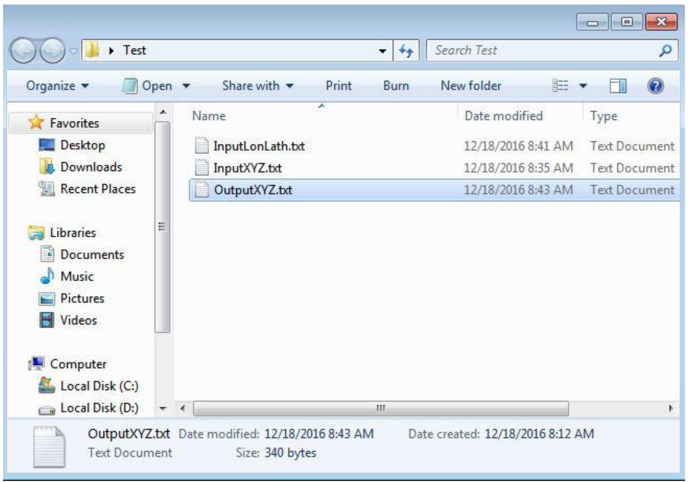

Figure (5.14): Result Dialog

10. Click on the OutputXYZ file & the result will appear.

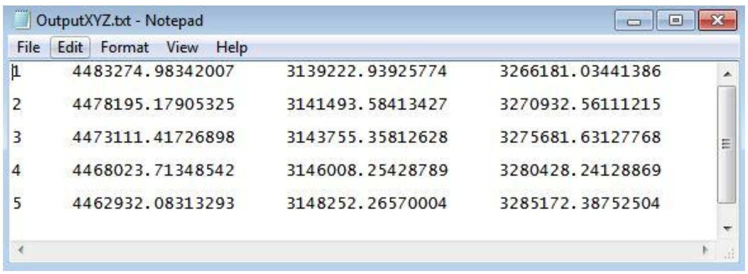

Figure (5.15): Example (5.5)

5.2.6. Transform From Geocentric to Geographic Coordinates for a File Points:

1. In input file select File Points.
2. Press X, Y, Z➔ lon,lat,h key.

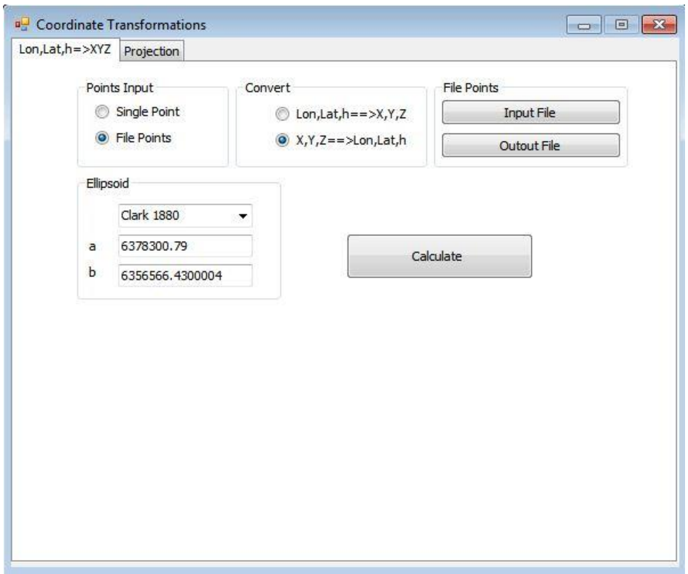

Figure (5.16): X,Y,Z➔lon,lat,h Dialog

3. Select Input File.
4. A new window will appear, select InputXYZ file, then open.

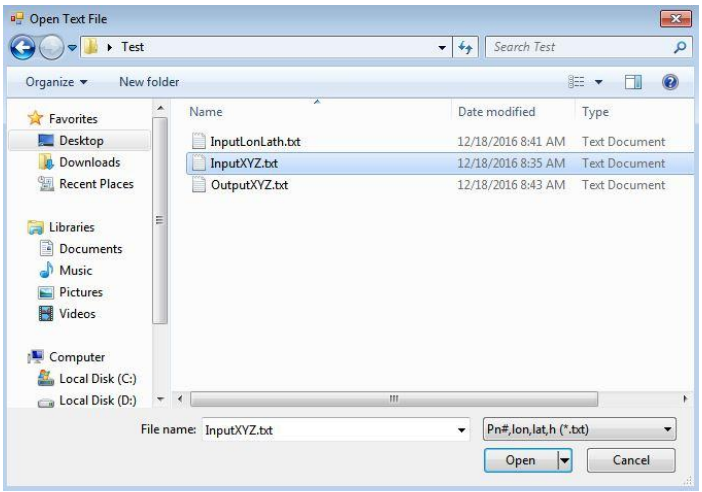

Figure (5.17): Open Dialog

5. After that will return to the main dialog, select the ellipsoid needed.

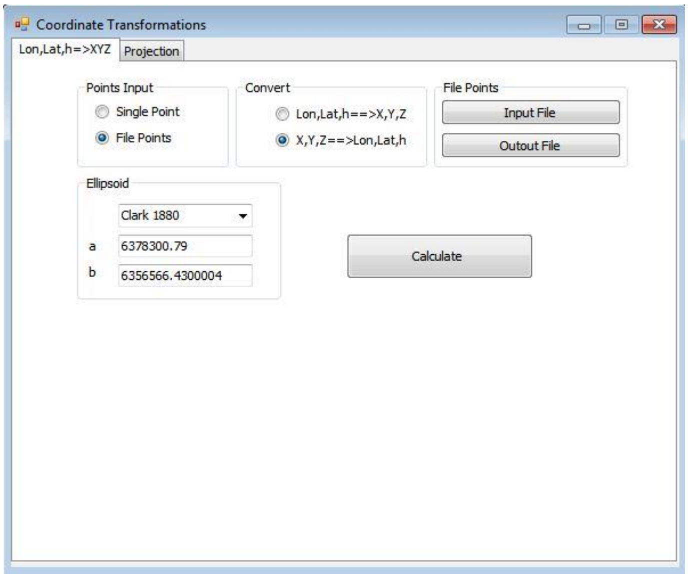

Figure (5.18): X,Y,Z➔lon,lat,h Dialog

6. Click select output file, & this window will appear. Write the file name to save (Outputlonlath) & then click save.

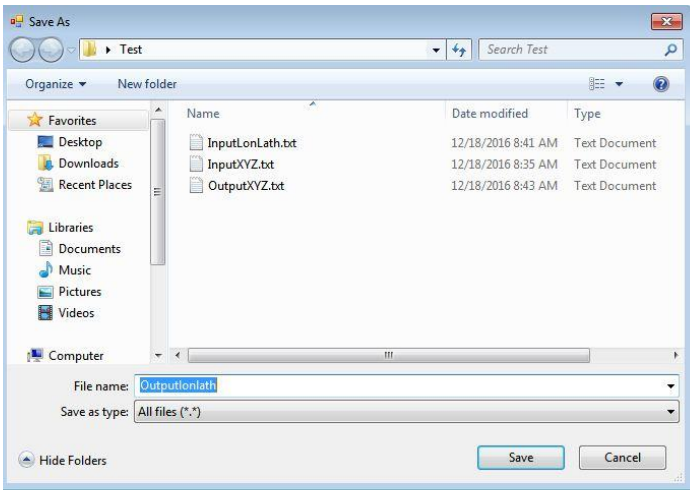

Figure (5.19): Save as Dialog

7. Now the result in the file that saved in will appear.

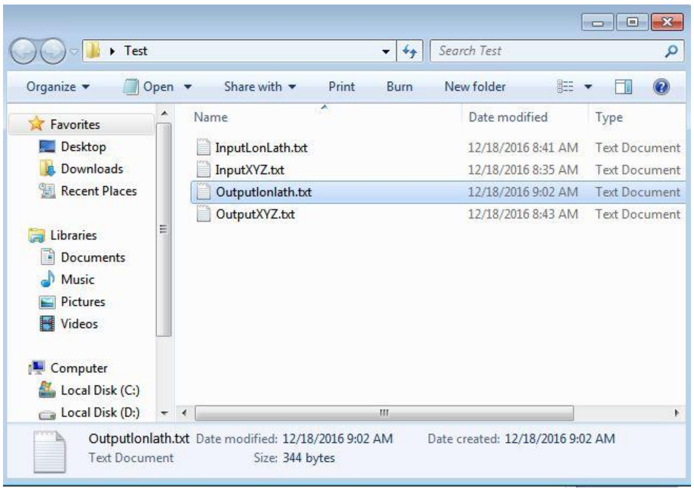

Figure (5.20): Result Dialog

10. Click on the Outputlonlath file & the result will appear.

Figure (5.21): Example (5.6)

5.2.7. Transform From Geographic to Projected Coordinates for a File Points:

1. In Points Input check File Points.
2. Press Direct Projection key.

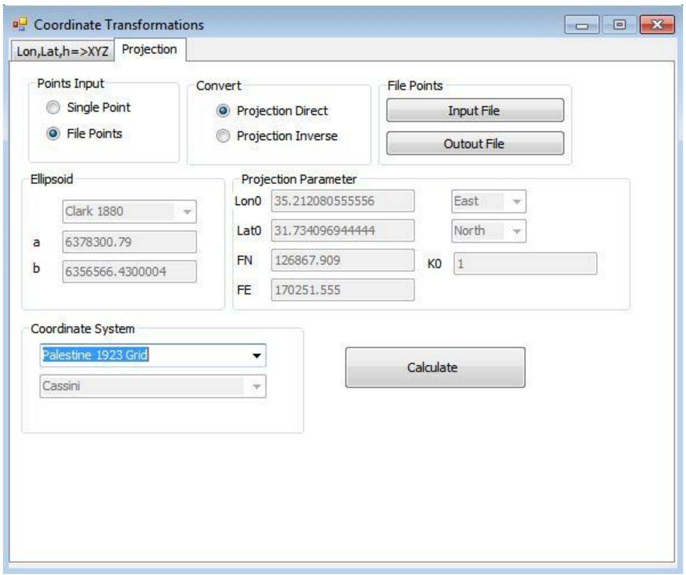

Figure (5.22): Direct Projection Dialog

3. Select Input File, then this window will appear.

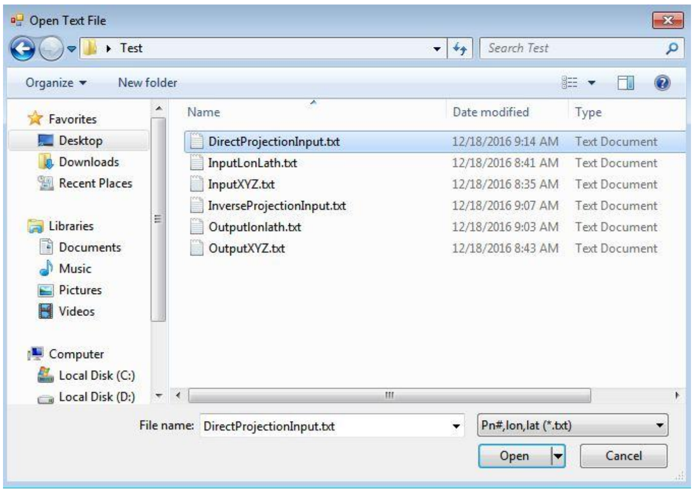

Figure (5.23): Open File Dialog

4. Choose (DirectProjectionInput) file & then open.

5. After that, return to the main dialog & then select the coordinate system needed & its projection.

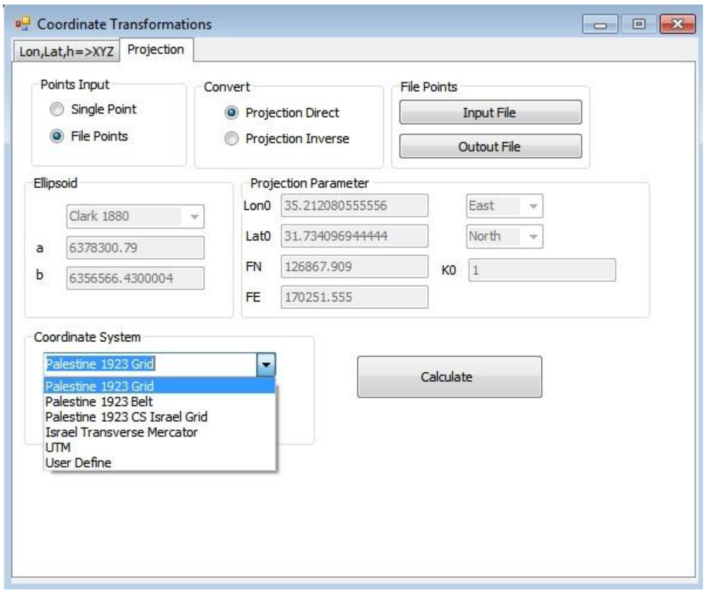

Figure (5.24): Direct Projection Dialog

6. The Ellipsoid will be selected automatically.

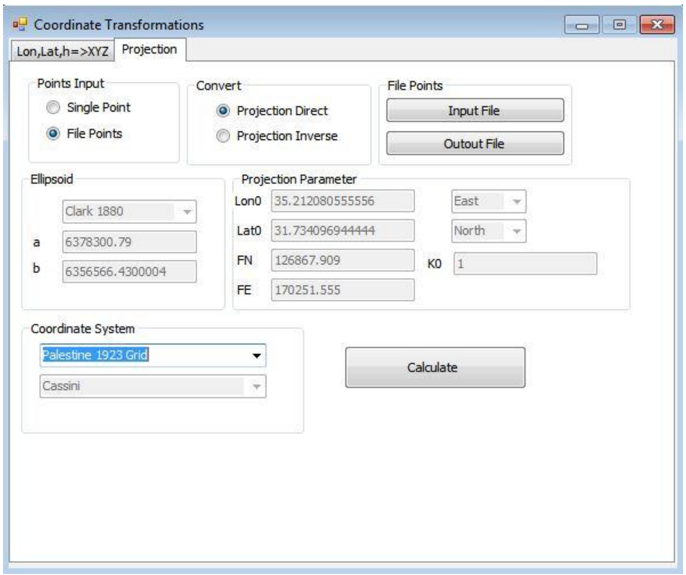

Figure (5.25): Direct Projection Dialog

7. Click select output file, & this window will appear. Write the file name to save (Output East North) & then click save.

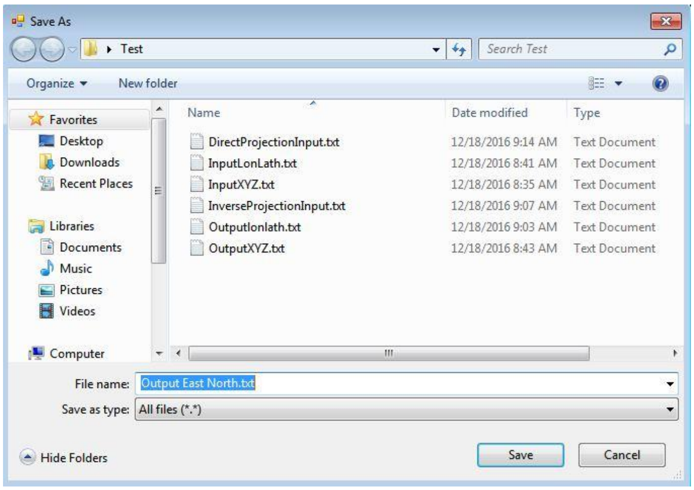

Figure (5.26): Save as Dialog

10. You can see the result from the file saved in.

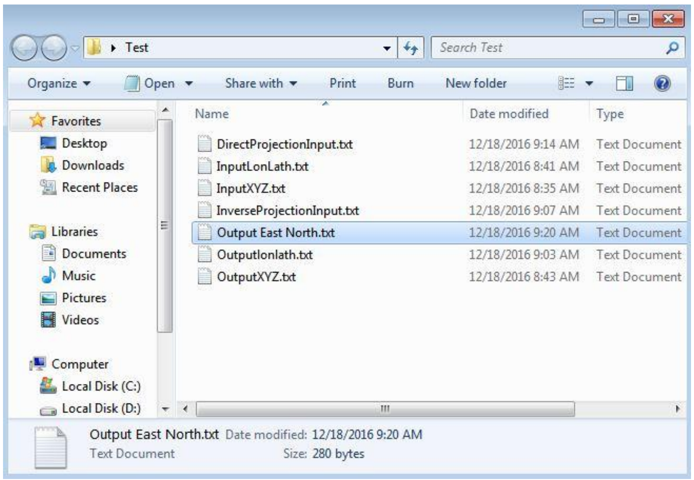

Figure (5.27): Result Dialog

11. Click on the Output East North file & the result will appear.

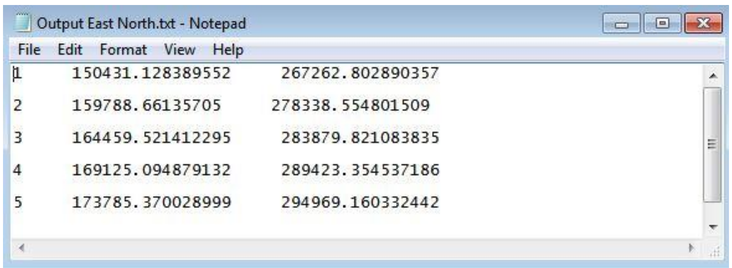

Figure (5.28): Example (5.7)

5.2.8. Transform From Projected to Geographic Coordinates for a File Points:

1. In Points Input check File Points.
2. Press Inverse Projection key.

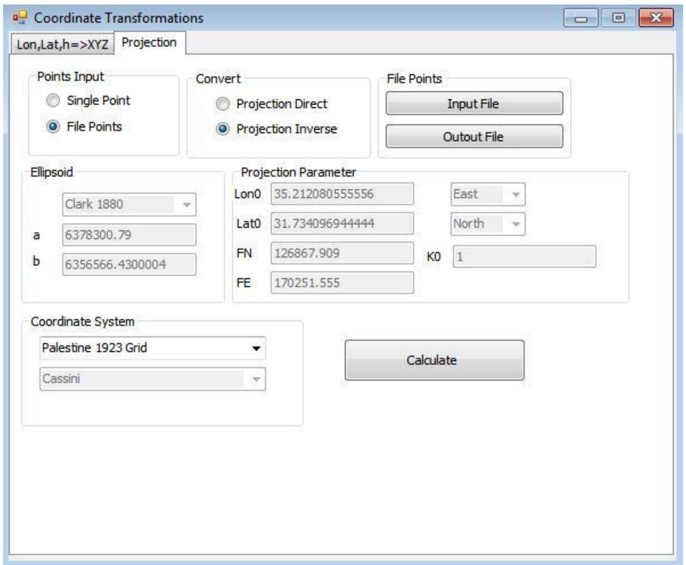

Figure (5.29): Inverse Projection Dialog

3. Select Input File, then this window will appear.

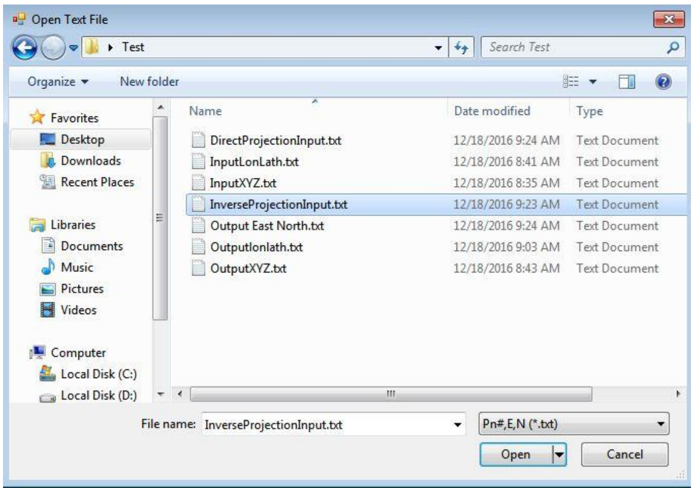

Figure (5.30): Open File Dialog

4. Choose (InverseProjectionInput) file & then open.
5. After that, return to the main dialog & then select the coordinate system needed & its projection.

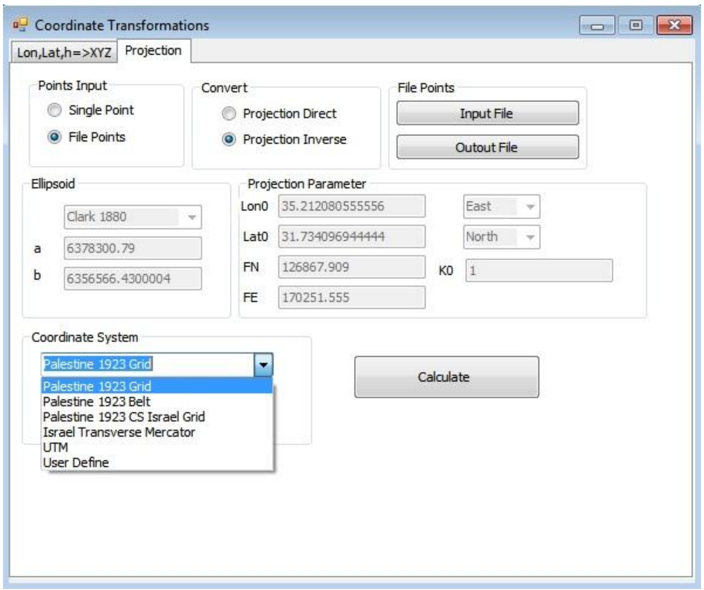

Figure (5.31): Inverse Projection Dialog

6. The Ellipsoid will be selected automatically.

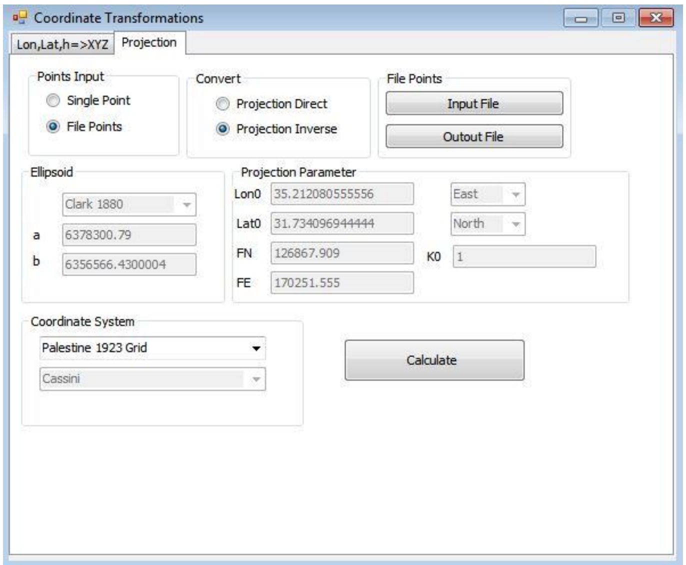

Figure (5.32): Inverse Projection Dialog

7. Click select output file, & this window will appear. Write the file name to save (Outputlonlat) & then click save.

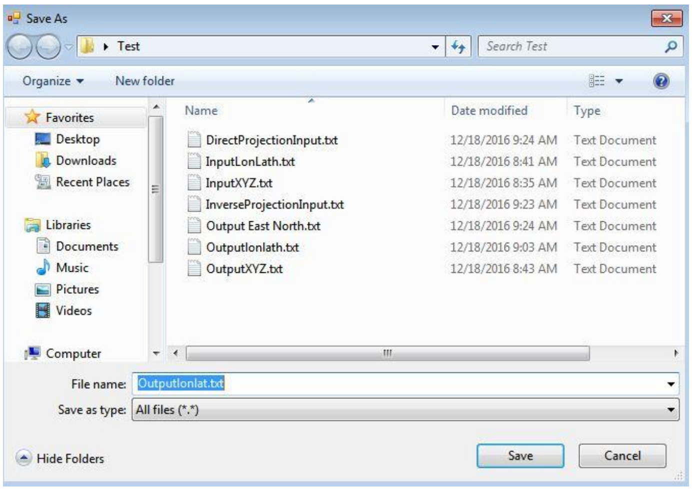

Figure (5.33): Save as Dialog

10. You can see the result from the file saved in.

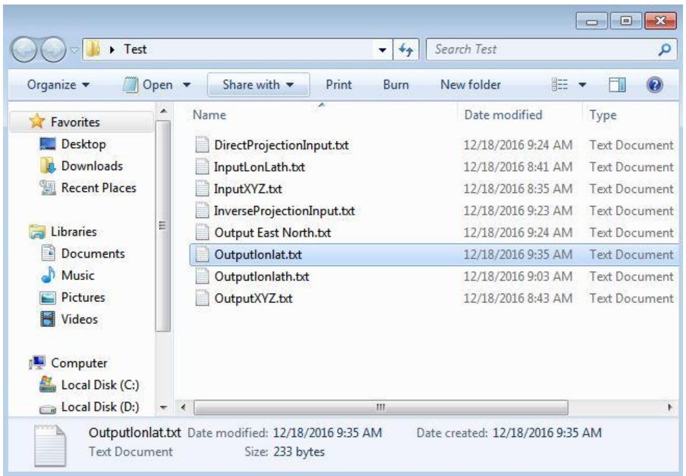

Figure (5.34): Result Dialog

11. Click on the Outputlonlat file & the result will appear.

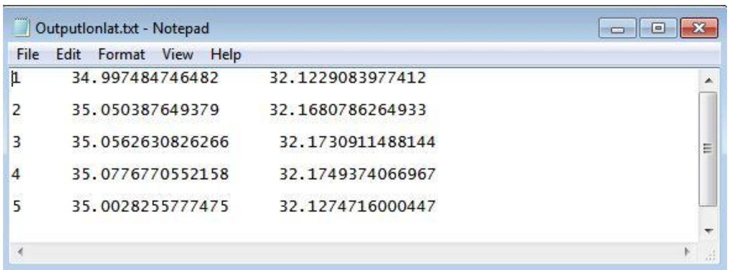

Figure (5.35): Example (5.8)

5.3. Using the Geo Transform 2.0 for Datum Transformations.

1. Open the Geo-Transform program.

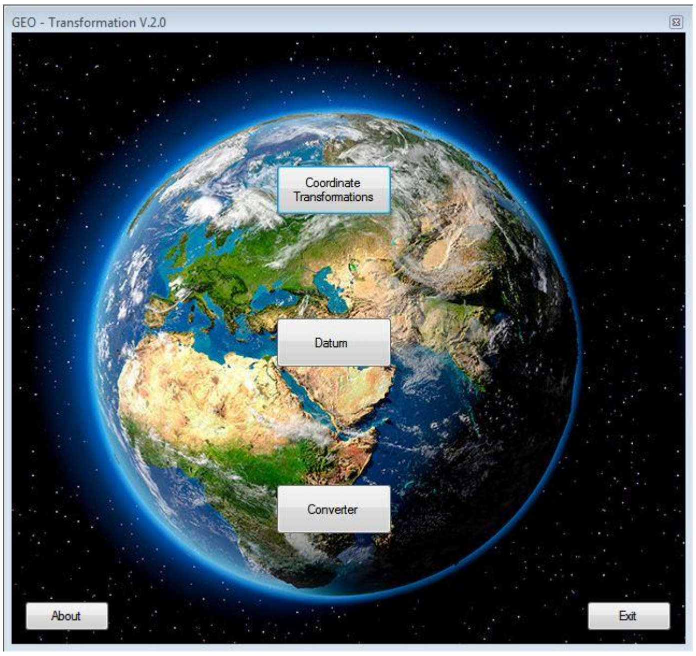

Figure (5.36): Geo-Transform Start Page

2. Click on Datum button.
3. The Datum Transformations window will appear as in figure (5.37).

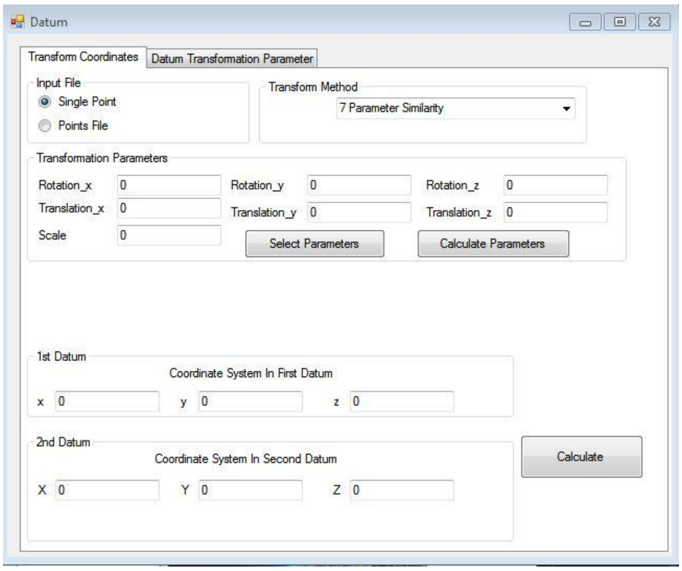

Figure (5.37): Datum Dialog

5.3.1. Calculating the Parameters:

To calculate the parameters there is two was:

1. From Transform Coordinates tab press, click on Calculate Parameters button.
2. Click on Datum Transformation Parameters tab.

In both ways the result will be the same, the following window will appear as in figure (5.38), so the user can use anyone.

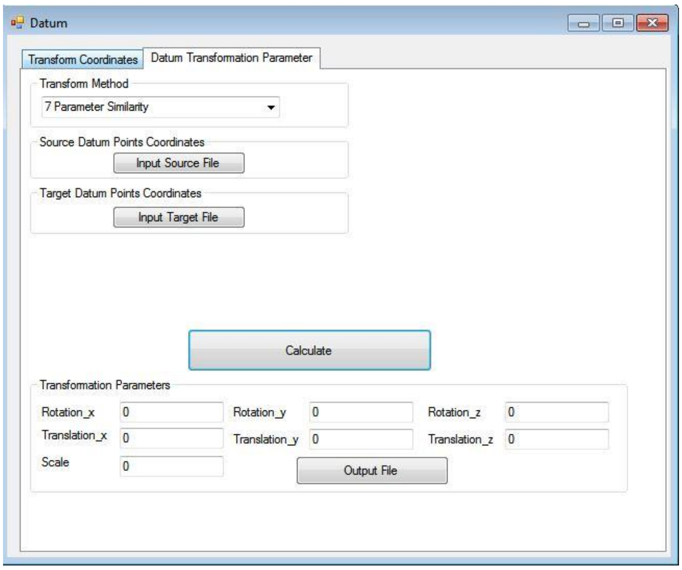

Figure (5.38): Datum Dialog

1. Select the transformation method.

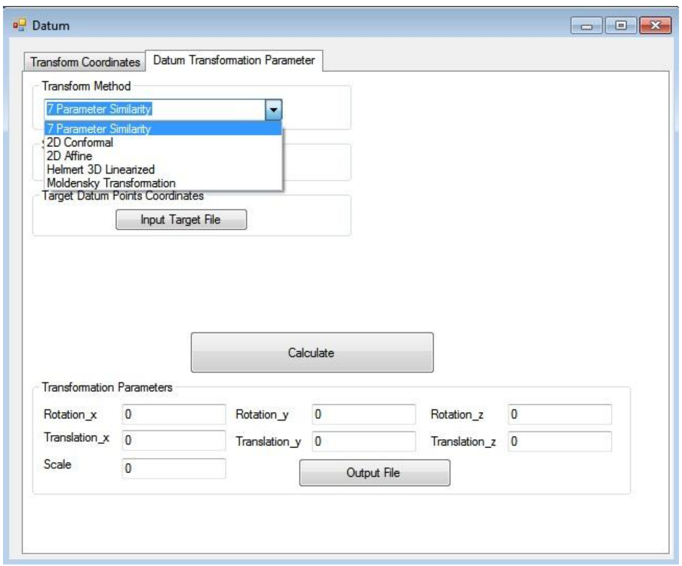

Figure (5.39): Datum Dialog

2. Press on Input Source File, then an open dialog will appear.
3. Choose the file with the source coordinate, then press open.

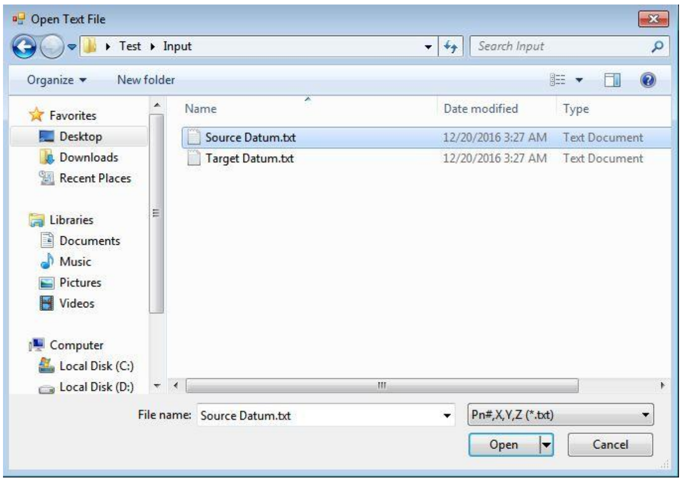

Figure (5.40): Open File Dialog

4. Figure (5.38) will show again, press on Input Target File, then an open dialog will appear.
5. Choose the file with target coordinates, then press open.

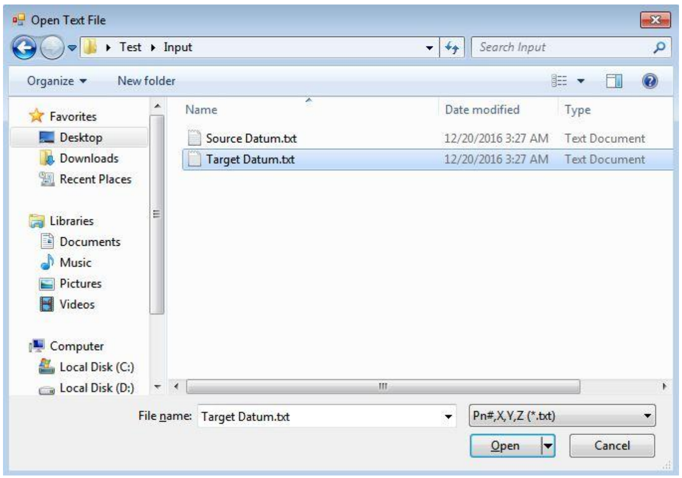

Figure (5.41): Open File Dialog

6. In Moldonskey transformation method, the user have to select the first & the second ellipsoid from the main dialog.

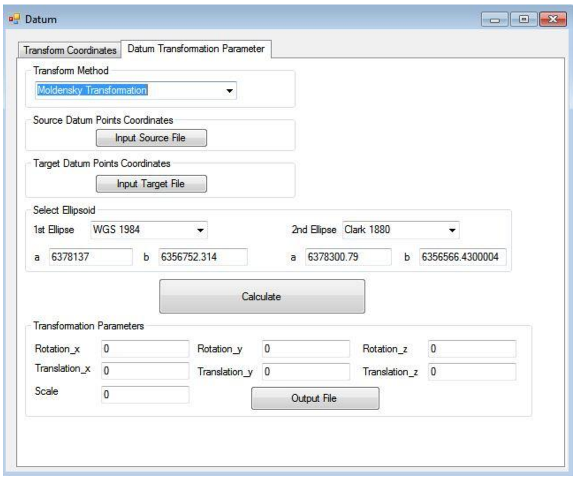

Figure (5.42): Datum Transformation Parameters Dialog

7. Now press calculate button from Datum Transformation Parameter dialog.
8. Click on Output File button to save the parameters in a file.

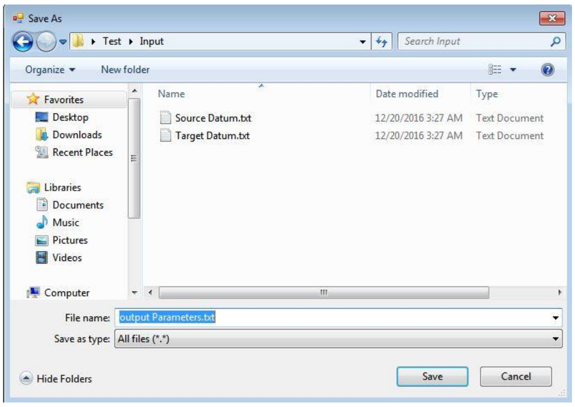

Figure (5.43): Save as Dialog

5.3.2. Transform using 7 Parameter for a Single Point:

1. In Points Input check Single Points.
2. In transformation method select 7 parameter similarity.

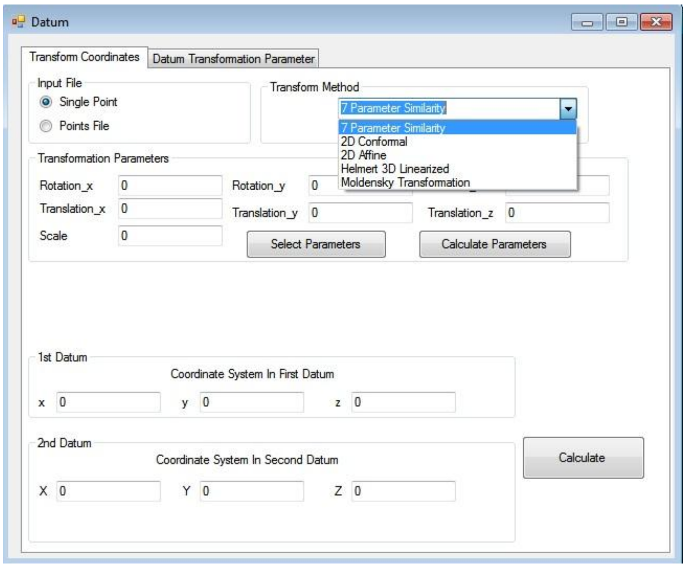

Figure (5.44): Datum Dialog

3. Enter the value of input data (x,y,z) in the first datum.

4. Enter the value of the seven parameters if exist, if not, calculate the parameters as shown in section 5.3.1.

5. Press calculate button

6. The result will appear as shown in figure (5.45).

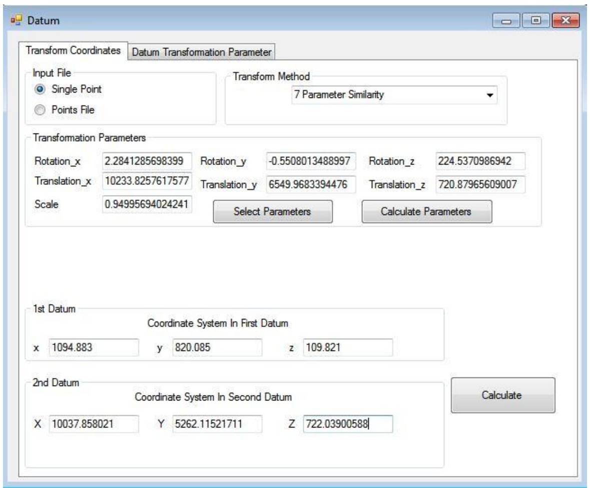

Figure (5.45): Example (5.9)

5.3.3. Transform using 2D Conformal for a Single Point:

1. In Points Input check Single Points.
2. In transformation method select 2D Conformal.

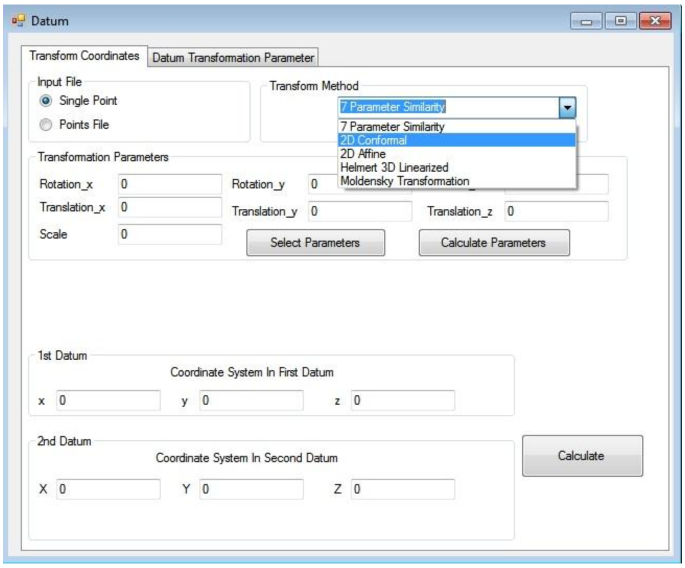

Figure (5.46): Datum Dialog

3. Enter the value of input data (x,y) in the first datum.
4. Enter the value of the four parameters if exist, if not, calculate the parameters as shown in section 5.3.1.
5. Press calculate button
6. The result will appear as shown in figure (5.47).

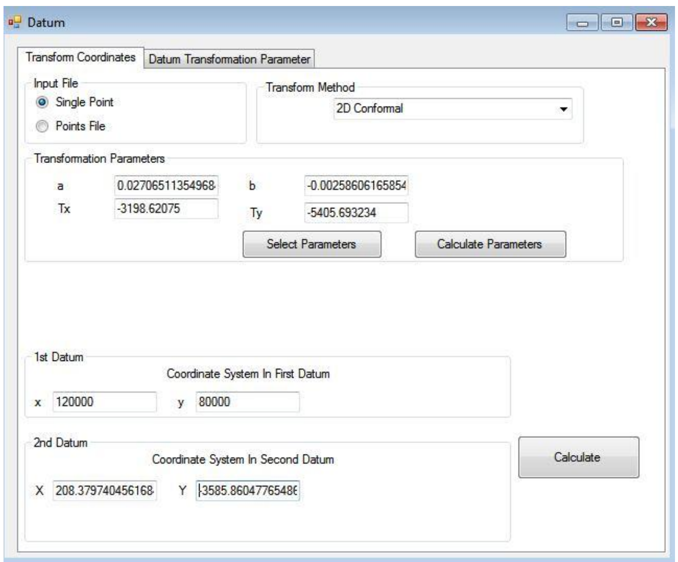

Figure (5.47): Example (5.10)

5.3.4. Transform using 2D Affine for a Single Point:

1. In Points Input check Single Points.
2. In transformation method select 2D Affine.

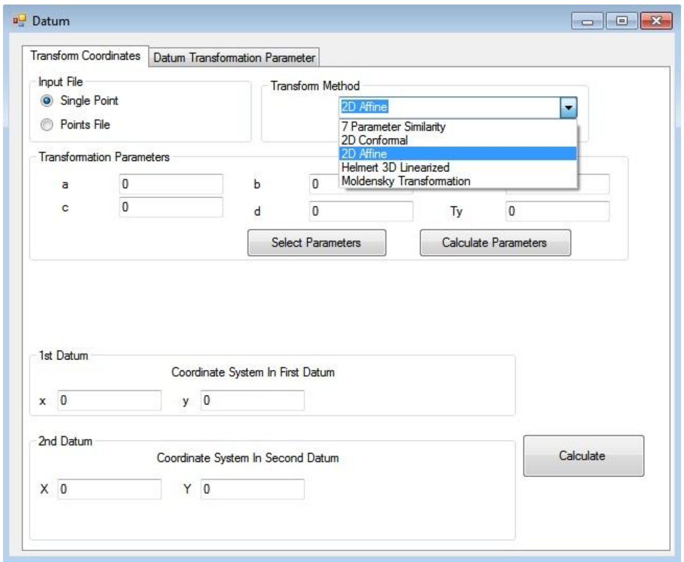

Figure (5.48): Datum Dialog

3. Enter the value of input data (x,y) in the first datum.
4. Enter the value of the six parameters if exist, if not, calculate the parameters as shown in section 5.3.1.
5. Press calculate button
6. The result will appear as shown in figure (5.49).

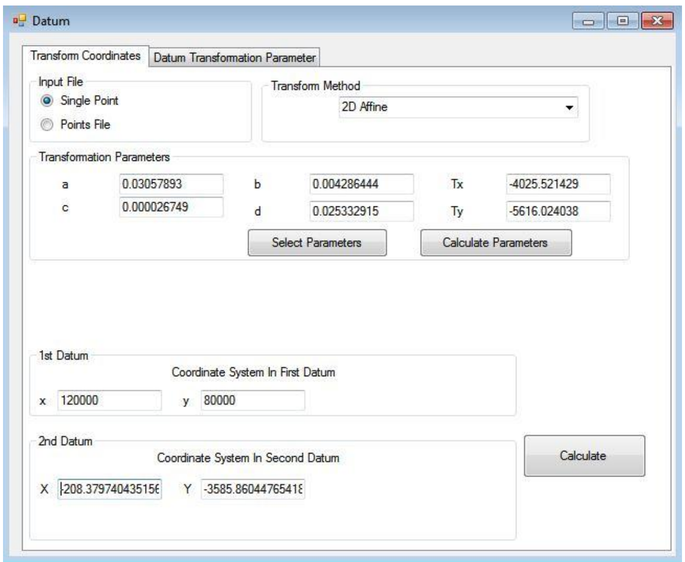

Figure (5.49): Example (5.11)

5.3.5. Transform using Helmert 3D Linearized for a Single Point:

1. In Points Input check Single Points.
2. In transformation method select Helmert 3D Linearized.

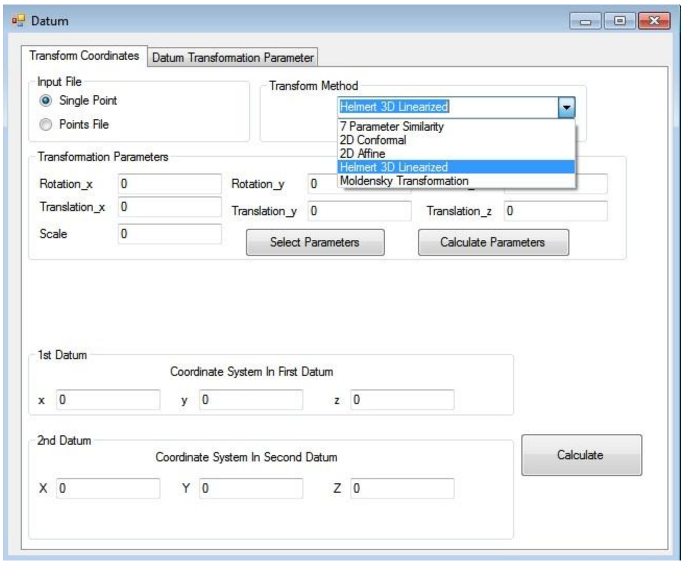

Figure (5.50): Datum Dialog

3. Enter the value of input data (x,y,z) in the first datum.
4. Enter the value of the seven parameters if exist, if not, calculate the parameters as shown in section 5.3.1.
5. Press calculate button
6. The result will appear as shown in figure (5.51).

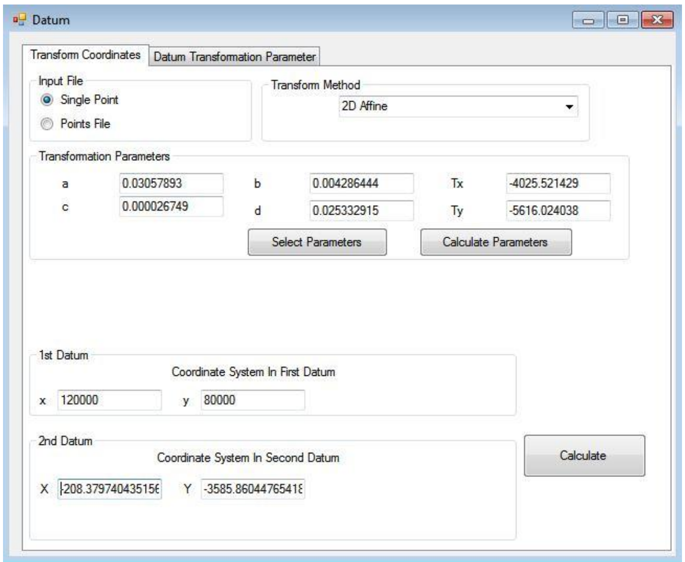

Figure (5.51): Example (5.12)

5.3.6. Transform using Molodensky Transformation for a Single Point:

1. In Points Input check Single Points.
2. In transformation method select Molodensky Transformation.

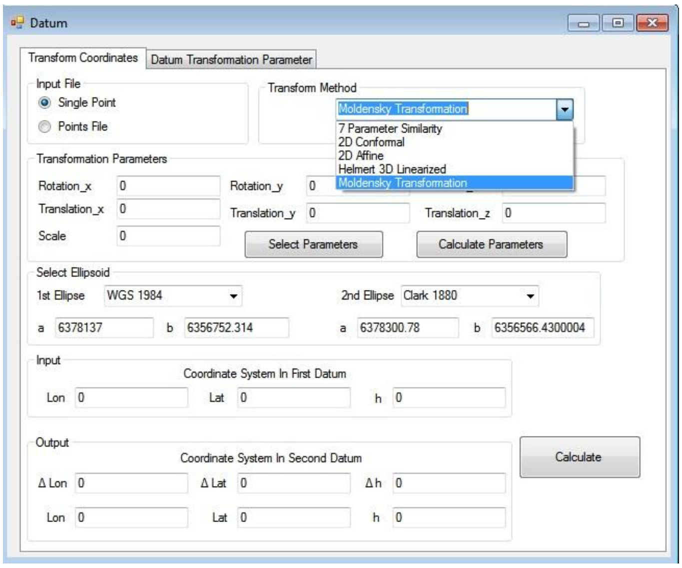

Figure (5.52): Datum Dialog

3. Enter the value of input data (x,y,z) in the first datum.
4. Enter the value of the seven parameters if exist, if not, calculate the parameters as shown in section 5.3.1.
5. Select ellipsoid for the first and the second datum.
5. Press calculate button
6. The result will appear as shown in figure (5.53).

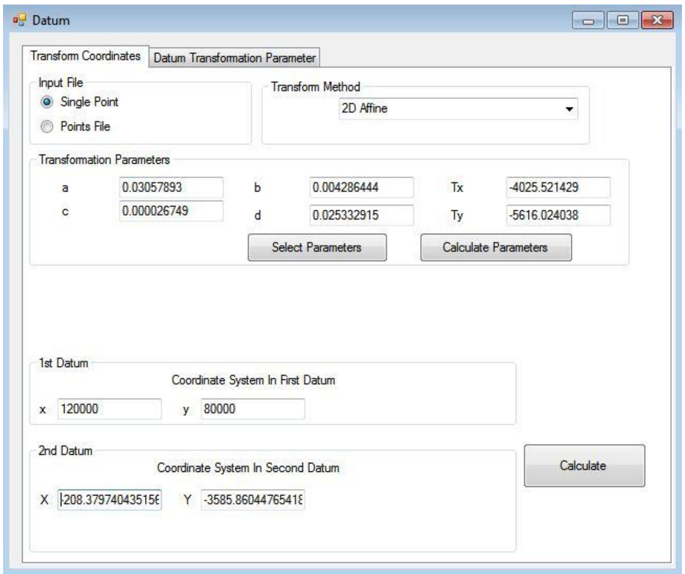

Figure (5.53): Example (5.13)

5.3.7. Transform using 7 Parameter for a Points File:

1. In Points Input check Points File.
2. In transformation method select 7 parameter similarity.

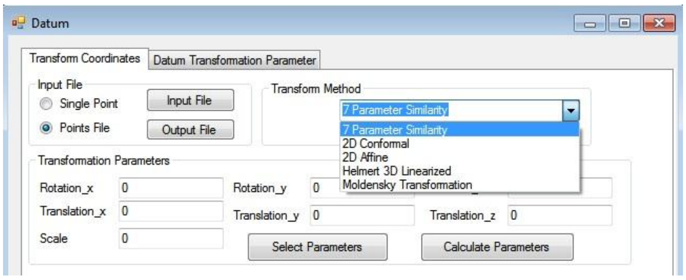

Figure (5.54): Datum Dialog

3. Click on the Input File button.

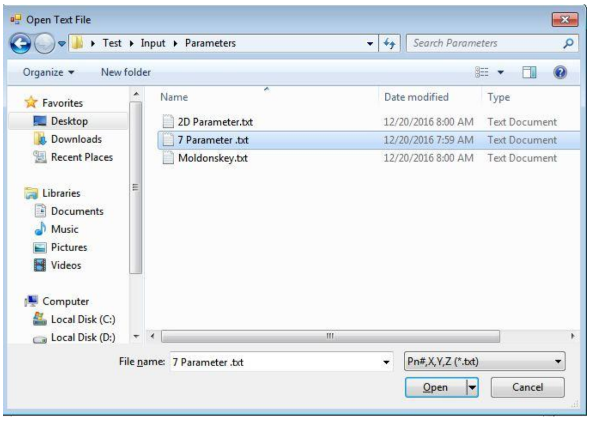

Figure (5.55): open Dialog

4. Choose 7 Parameter file, then press open.
5. Enter the value of the parameters if exist, if not, calculate the parameters as shown in section 5.3.1.
5. In Transform Coordinates window click on the Output File button.

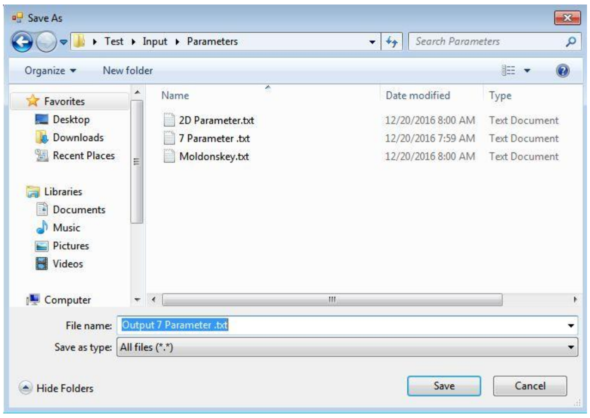

Figure (5.56): open Dialog

6. Enter the name of the output file as shown in figure (5.55).
7. The result will appear as shown in figure (5.56).

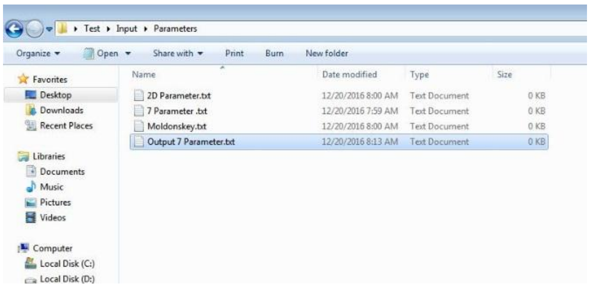

Figure (5.57): Example (5.14)

5.3.8. Transform using 2D Conformal for a Points File:

1. In Points Input check Points File.
2. In transformation method select 2D Conformal.

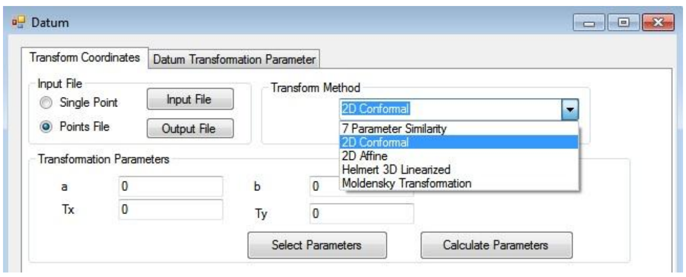

Figure (5.58): Datum Dialog

3. Click on the Input File button.

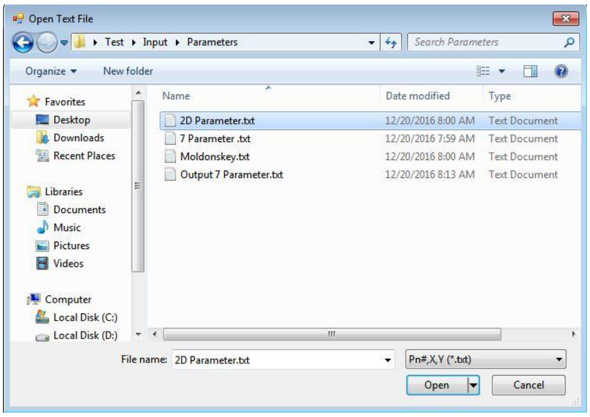

Figure (5.59): open Dialog

4. Choose 2D Parameter file, then press open.

5. Enter the value of the parameters if exist, if not, calculate the parameters as shown in section 5.3.1.

6. In Transform Coordinates window click on the Output File button.

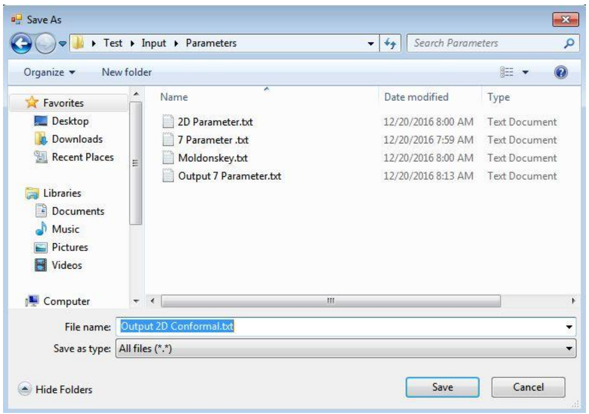

Figure (5.60): open Dialog

6. Enter the name of the output file as shown in figure (5.59).

7. The result will appear as shown in figure (5.60).

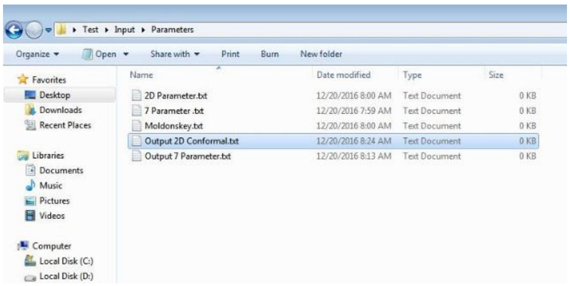

Figure (5.61): Example (5.15)

5.3.9. Transform using 2D Affine for a Points File:

1. In Points Input check Points File.
2. In transformation method select 2D Affine.

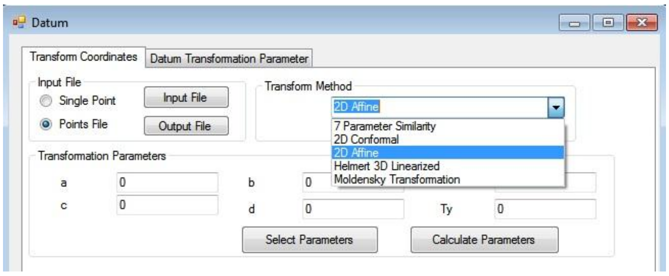

Figure (5.62): Datum Dialog

3. Click on the Input File button.

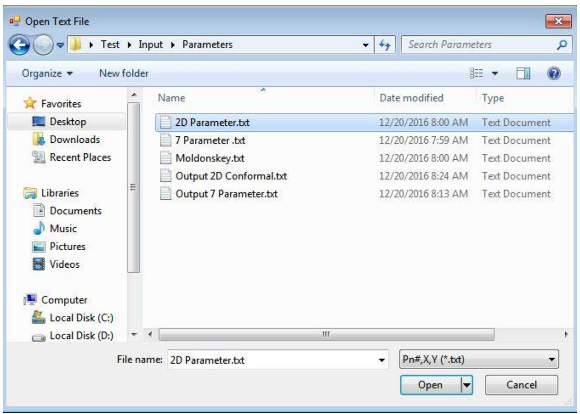

Figure (5.63): open Dialog

4. Choose 2D Parameter file, then press open.

5. Enter the value of the parameters if exist, if not, calculate the parameters as shown in section 5.3.1.

6. In Transform Coordinates window click on the Output File button.

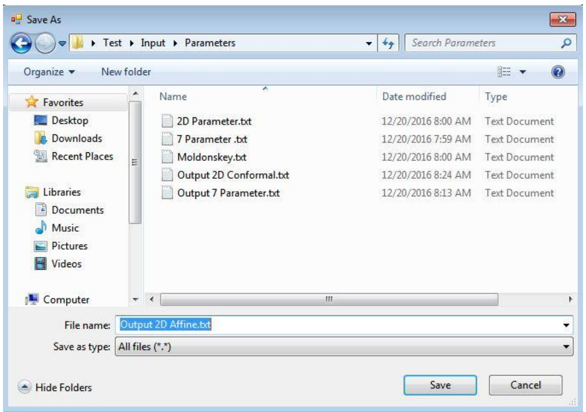

Figure (5.64): open Dialog

6. Enter the name of the output file as shown in figure (5.59).

7. The result will appear as shown in figure (5.60).

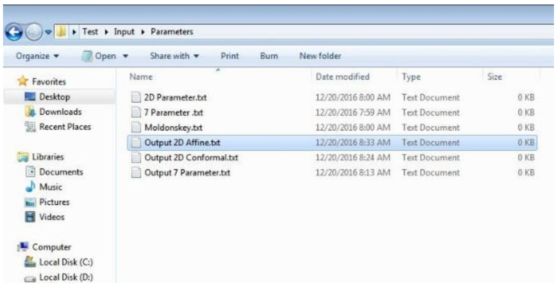

Figure (5.65): Example (5.16)

5.3.10. Transform using Helmert for a Points File:

1. In Points Input check Points File.

2. In transformation method select Helmert.

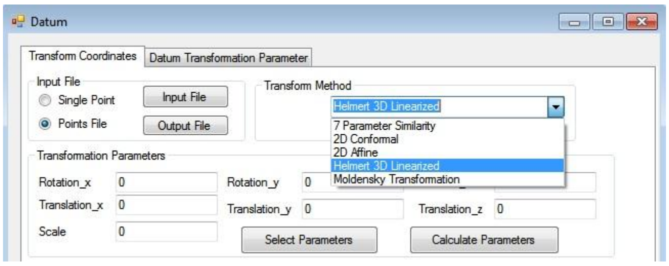

Figure (5.66): Datum Dialog

3. Click on the Input File button.

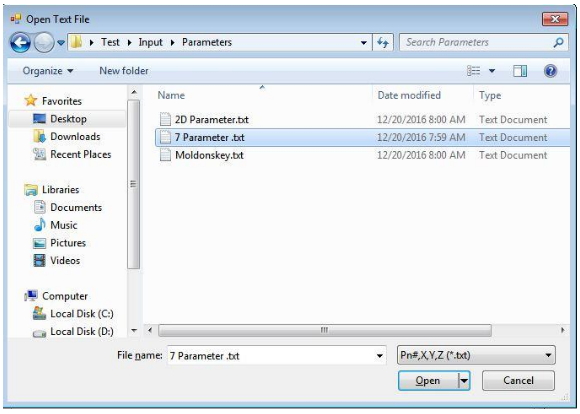

Figure (5.67): open Dialog

4. Choose 7 Parameter file, then press open.

5. Enter the value of the parameters if exist, if not, calculate the parameters as shown in section 5.3.1.

6. In Transform Coordinates window click on the Output File button.

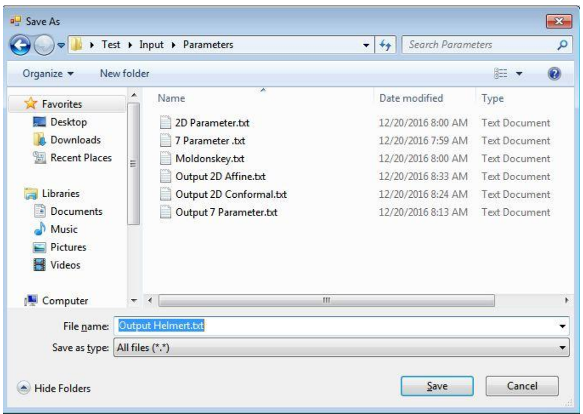

Figure (5.68): open Dialog

6. Enter the name of the output file as shown in figure (5.67).
7. The result will appear as shown in figure (5.68).

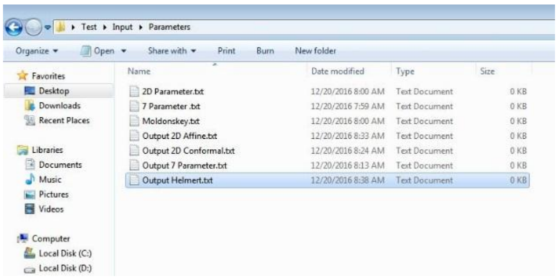

Figure (5.69): Example (5.17)

5.3.11. Transform using Molodensky for a Points File:

1. In Points Input check Points File.
2. In transformation method select Molodensky.

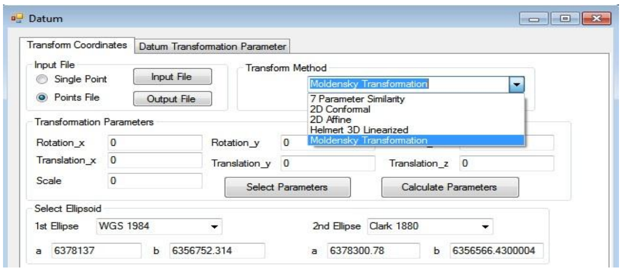

Figure (5.70): Datum Dialog

3. Click on the Input File button.

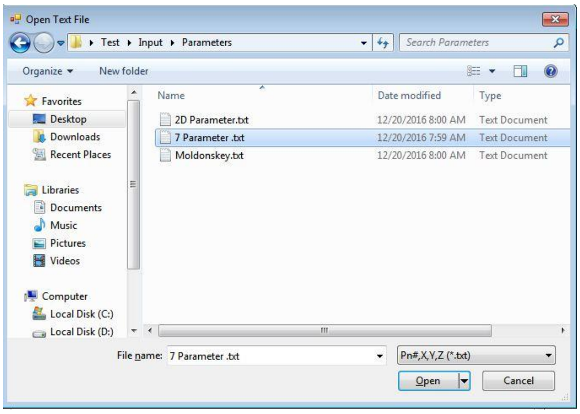

Figure (5.71): open Dialog

4. Choose Molodensky file, then press open.

5. Enter the value of the parameters if exist, if not, calculate the parameters as shown in section 5.3.1.

6. In Transform Coordinates window click on the Output File button.

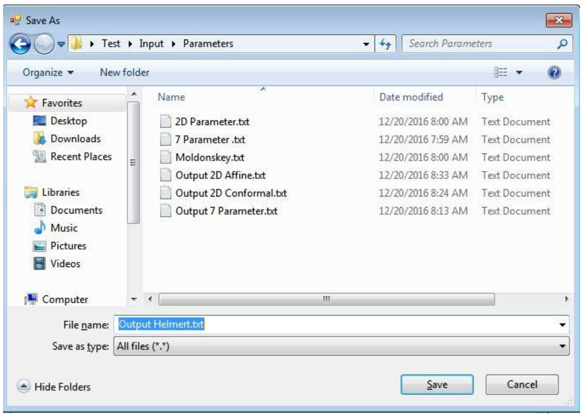

Figure (5.72): open Dialog

6. Enter the name of the output file as shown in figure (5.67).

7. The result will appear as shown in figure (5.68).

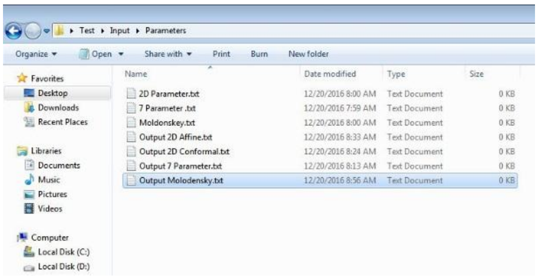

Figure (5.73): Example (5.18)

5.4. Using the Geo Transform for Angular Conversion.

1. Open the Geo-Transform program.

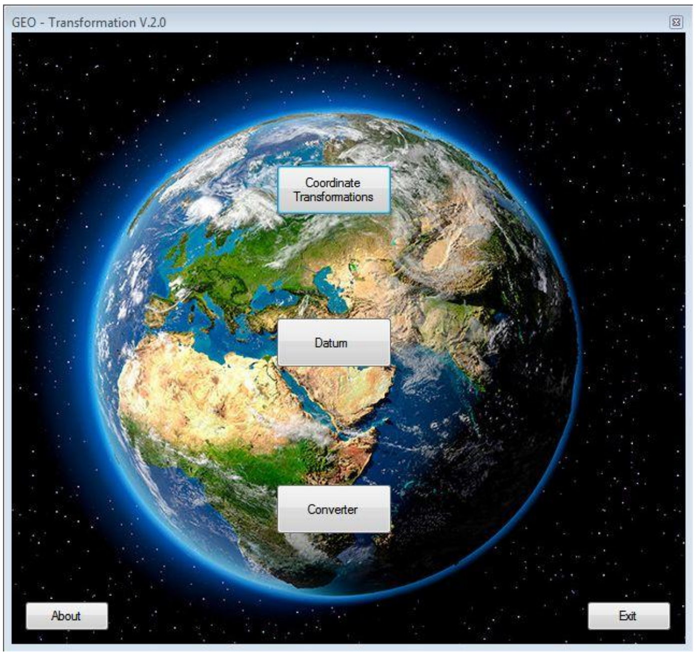

Figure (5.74): Geo-Transform Start Page

2. Click on Converter button.
3. The Converter window will appear as in figure (5.75).

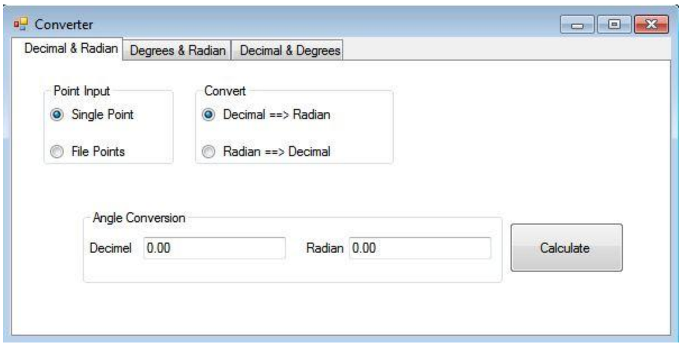

Figure (5.75): Datum Dialog

5.4.1. Convert From Decimal Degrees to Radian for a Single Point:

1. Check the Single Point key.
2. Check the Decimal➔Radian key.
3. Input the value of the angle in decimals

4. Press the Calculate button.
5. The result will be shown as in figure (5.76)

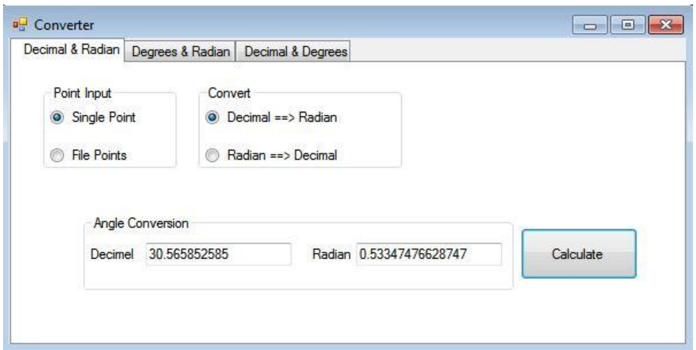

Figure (5.76): Converter dialog

5.4.2. Convert From Radian to Decimal Degree for a Single Point:

1. Check the Single Point key.
2. Check the Radian➔Decimal key.
3. Input the value of the angle in Radian.
4. Press the Calculate button.
5. The result will be shown as in figure (5.77)

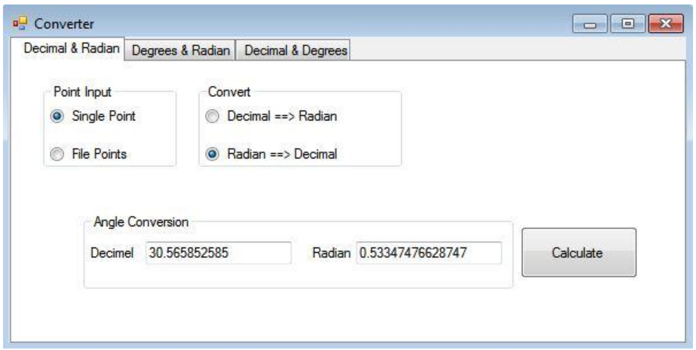

Figure (5.77): Converter dialog

5.4.3. Convert From Degrees\Minutes\Seconds to Radian for a Single Point:

1. Check the Single Point key.
2. Click on the Degrees & Radian tab.

3. Check the Degrees➔Radian key.
4. Input the value of the angle in Degrees\Minutes\Seconds.
5. Press the Calculate button.
6. The result will be shown as in figure (5.78)

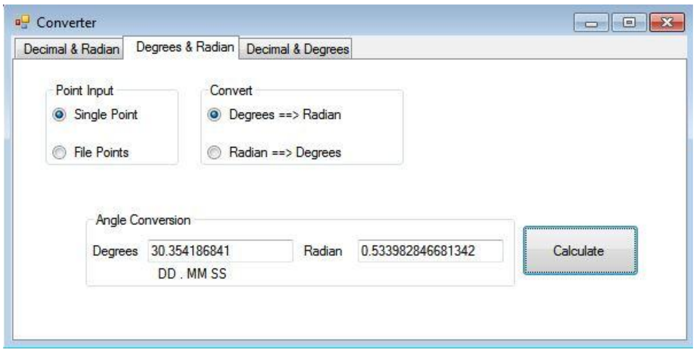

Figure (5.78): Converter dialog

.

5.4.4. Convert From Radian to Degrees\Minutes\Seconds for a Single Point:

1. Check the Single Point key.
2. Click on the Degrees & Radian tab.
3. Check the Radian➔Degrees key.
4. Input the value of the angle in Radian.
5. Press the Calculate button.
6. The result will be shown as in figure (5.79)

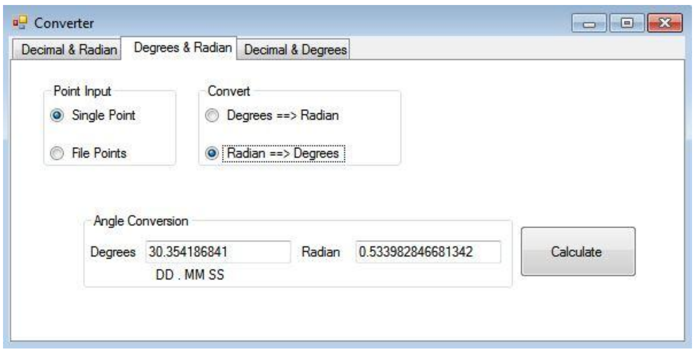

Figure (5.79): Converter dialog

5.4.5. Convert From Decimal to Degrees\Minutes\Seconds for a Single Point:

1. Check the Single Point key.
2. Click on the Decimal & Degrees tab.
3. Check the Decimal➔Degrees key.
4. Input the value of the angle in Decimal.
5. Press the Calculate button.
6. The result will be shown as in figure (5.80)

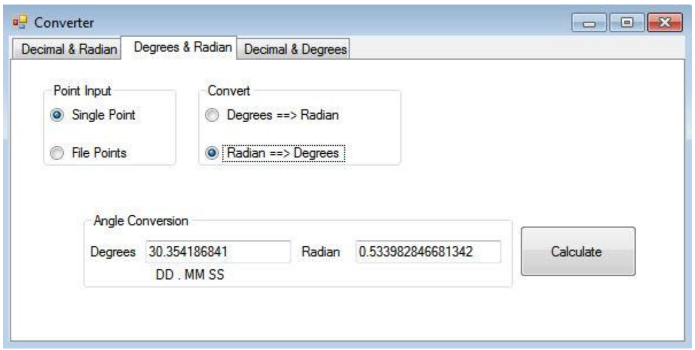

Figure (5.80): Converter dialog

5.4.6. Convert From Degrees\Minutes\Seconds to Decimal for a Single Point:

1. Check the Single Point key.
2. Click on the Decimal & Degrees tab.
3. Check the Degrees➔Decimal key.
4. Input the value of the angle in Degrees\Minutes\Seconds.
5. Press the Calculate button.
6. The result will be shown as in figure (5.81)

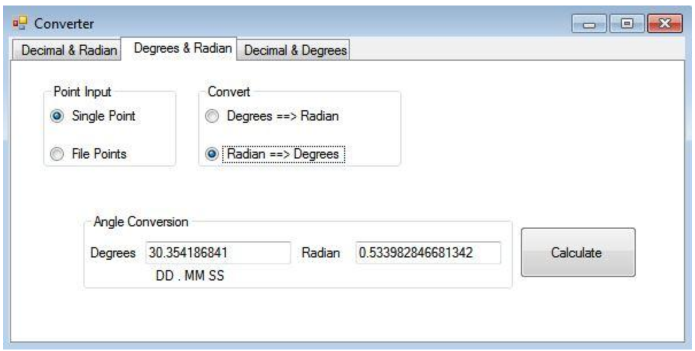

Figure (5.81): Converter dialog

5.4.7. Convert From Decimal Degrees to Radian for a Points File:

1. Check the File Points key.
2. Check the Decimal➔Radian key.
3. Determine the number of the columns in the file want to convert.
4. Click on the Input button.

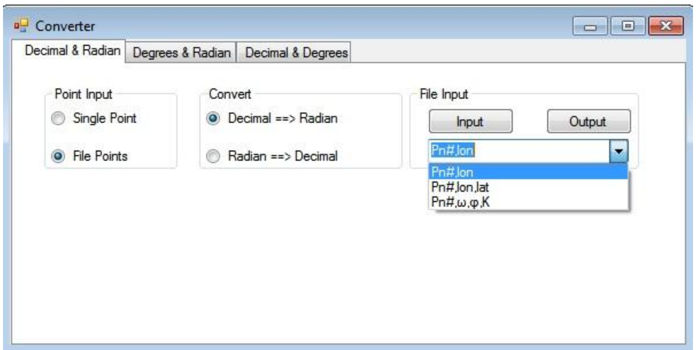

Figure (5.82): Converter dialog

5. An open dialog will show, choose the (dd to rad) file.

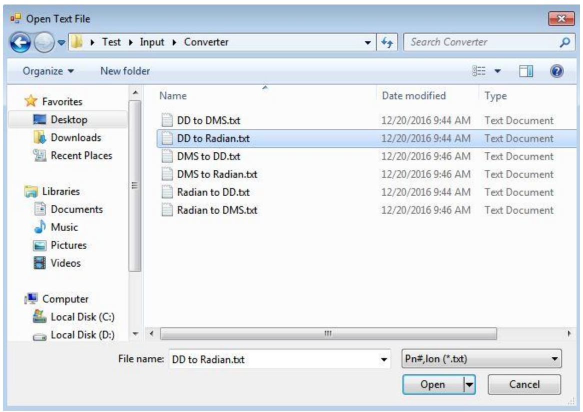

Figure (5.83): open dialog

6. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

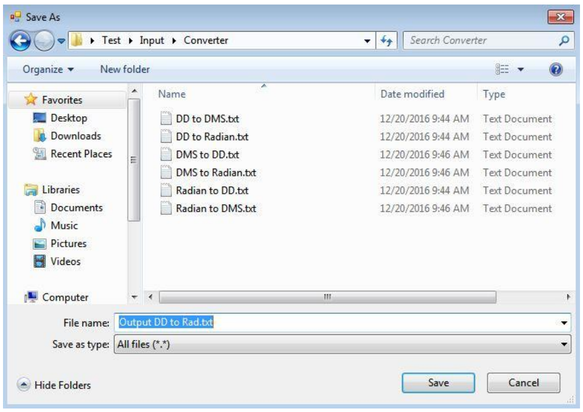

Figure (5.84): Save as dialog

7. The result will appear as shown in figure (5.85).

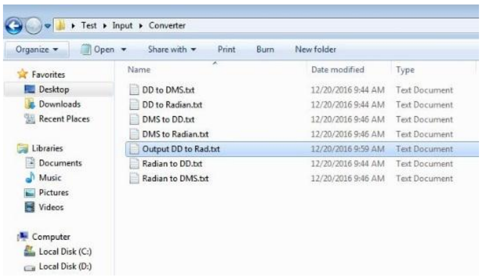

Figure (5.85): Example (5.19)

5.4.8. Convert From Radian to Decimal Degrees for a Points File:

1. Click on the Decimal & Radian tab.
2. Check the File Points key.
3. Check the Radian➔Decimal key.
4. Determine the number of the columns in the file want to convert.
5. Click on the Input button.

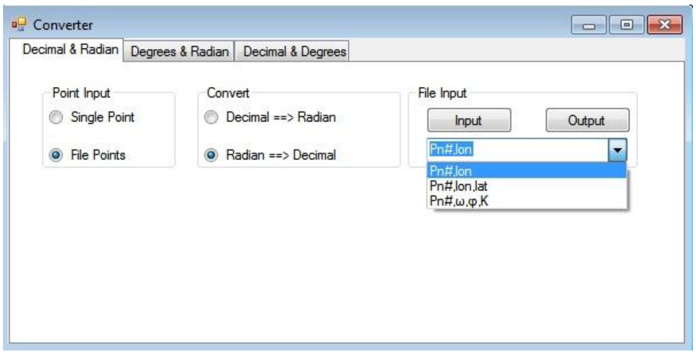

Figure (5.86): Converter dialog

6. An open dialog will show, choose the (rad to dd) file.

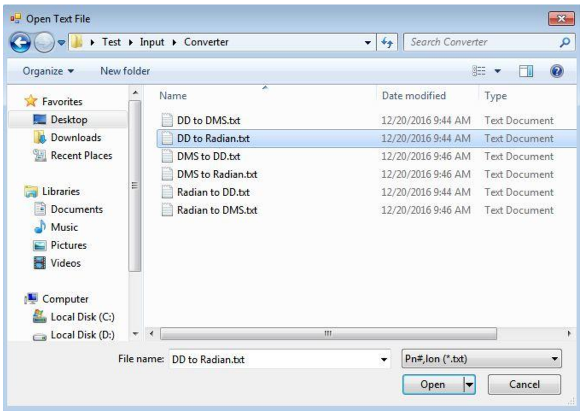

Figure (5.87): open dialog

7. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

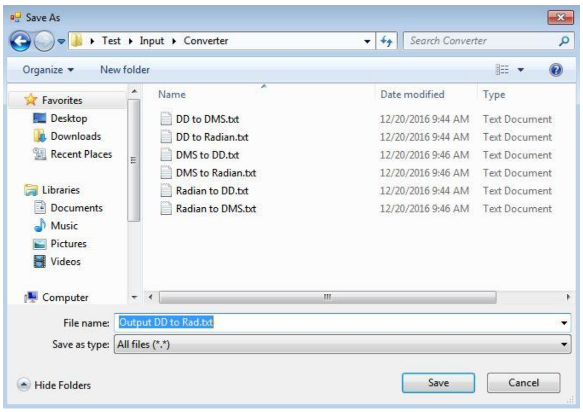

Figure (5.88): Save as dialog

8. The result will appear as shown in figure (5.89).

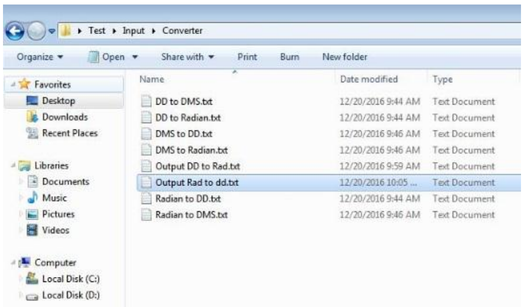

Figure (5.89): Example (5.20)

5.4.9. Convert From Degrees\Minutes\Seconds to Radian for a Points File:

1. Click on the Degrees & Radian tab.
2. Check the File Points key.
3. Check the Degrees➔Radian key.
4. Determine the number of the columns in the file want to convert.
5. Click on the Input button.

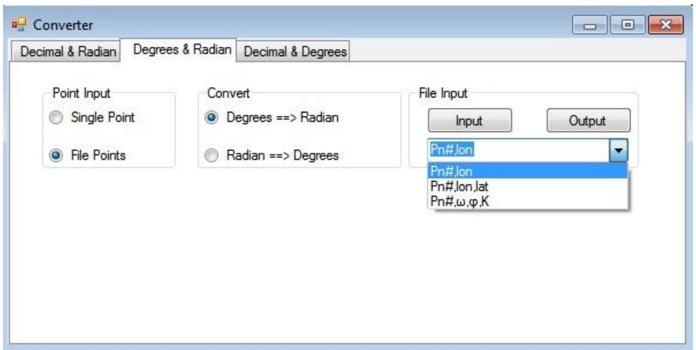

Figure (5.90): Converter dialog

6. An open dialog will show, choose the (dms to rad) file.

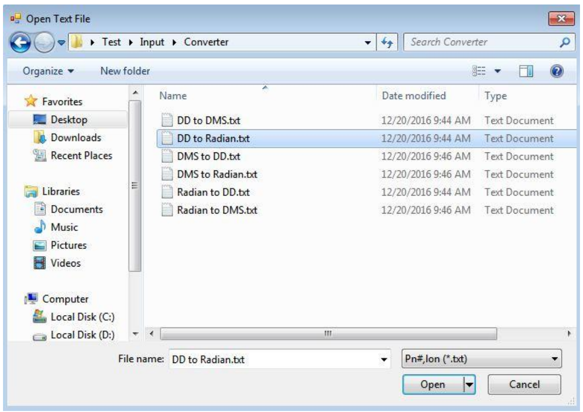

Figure (5.91): open dialog

7. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

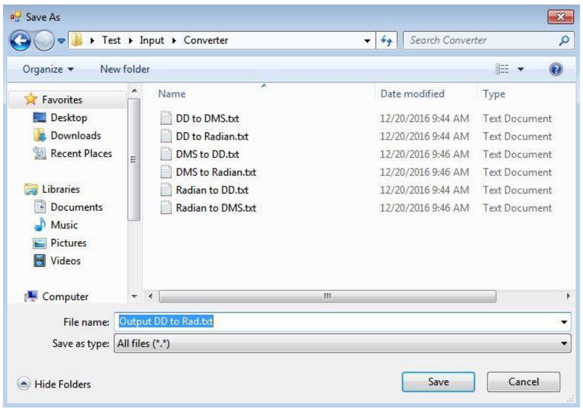

Figure (5.92): Save as dialog

8. The result will appear as shown in figure (5.92).

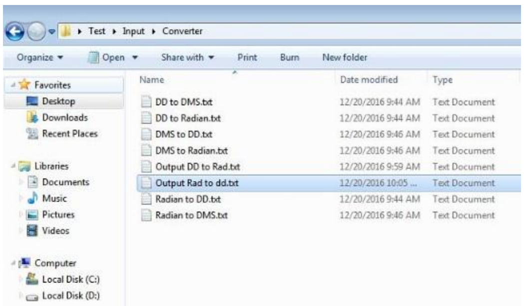

Figure (5.93): Example (5.21)

5.4.10. Convert From Radian to Degrees\Minutes\Seconds for a Points File:

1. Click on the Degrees & Radian tab.
2. Check the File Points key.
3. Check the Radian➔Degrees key.
4. Determine the number of the columns in the file want to convert.
5. Click on the Input button.

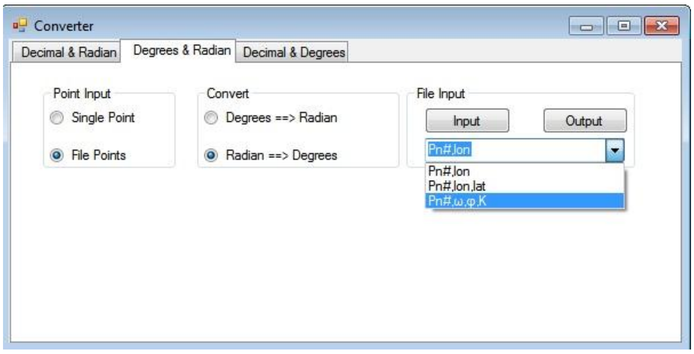

Figure (5.94): Converter dialog

6. An open dialog will show, choose the (rad to dms) file.

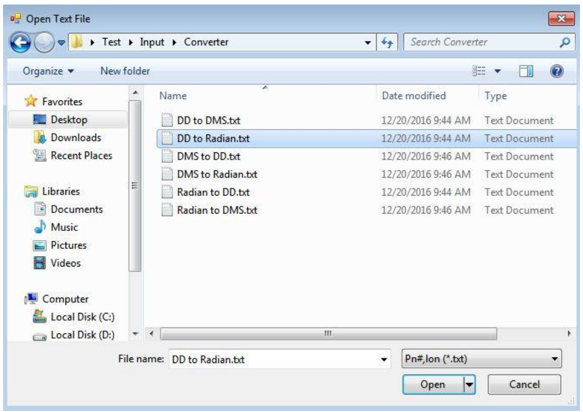

Figure (5.95): open dialog

7. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

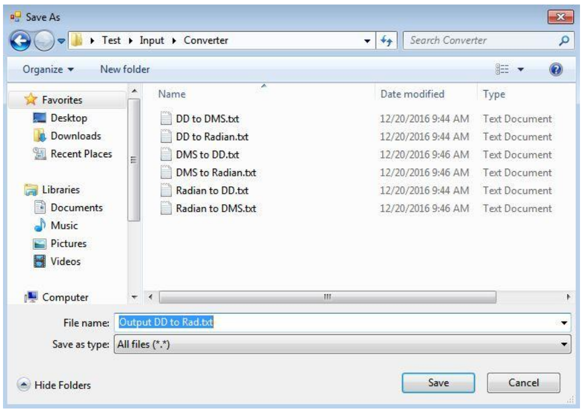

Figure (5.96): Save as dialog

8. The result will appear as shown in figure (5.97).

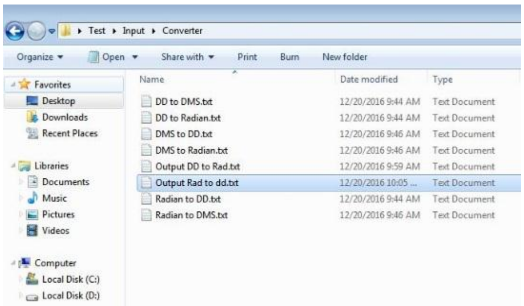

Figure (5.97): Example (5.22)

5.4.11. Convert From Decimal to Degrees\Minutes\Seconds for a Points File:

1. Click on the Decimal & Degrees tab.
2. Check the File Points key.
3. Check the Decimal➔Degrees key.
4. Determine the number of the columns in the file want to convert.
5. Click on the Input button.

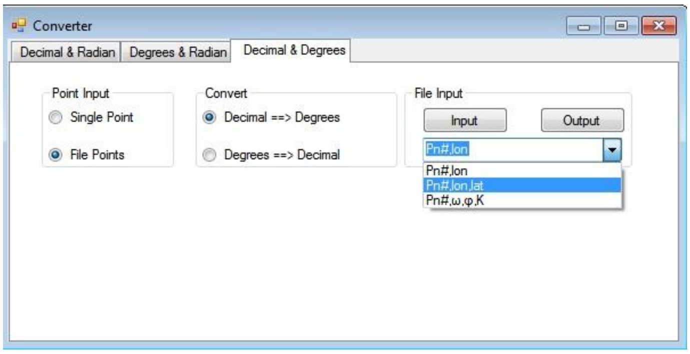

Figure (5.98): Converter dialog

6. An open dialog will show, choose the (dd to dms) file.

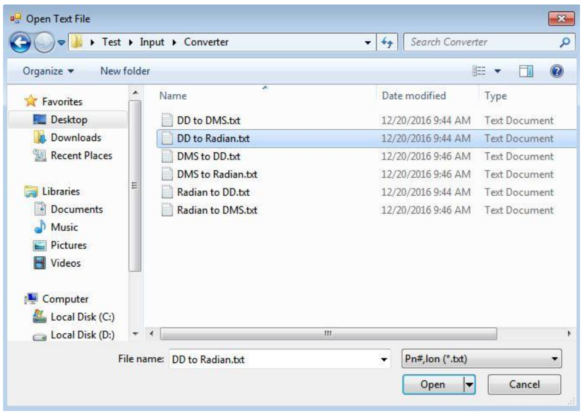

Figure (5.99): open dialog

7. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

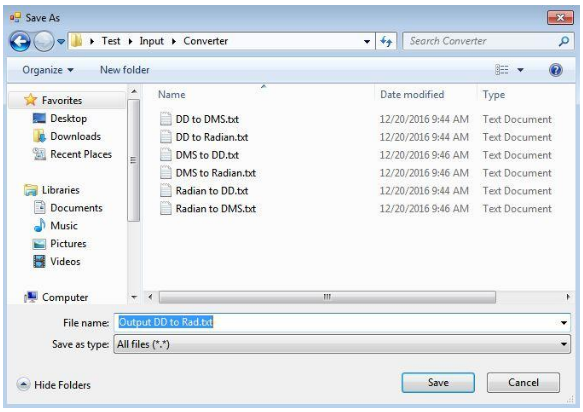

Figure (5.100): Save as dialog

8. The result will appear as shown in figure (5.101).

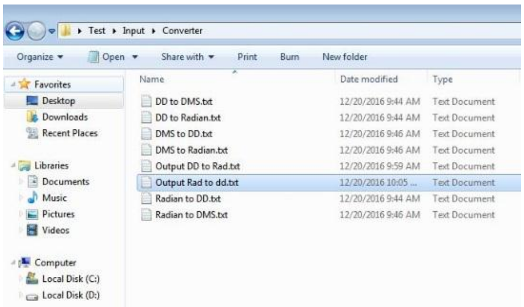

Figure (5.101): Example (5.23)

5.4.12. Convert From Degrees\Minutes\Seconds to Decimal for a Points File:

1. Click on the Decimal & Degrees tab.
2. Check the File Points key.
3. Check the Degrees➔Decimal key.
4. Determine the number of the columns in the file want to convert.
5. Click on the Input button.

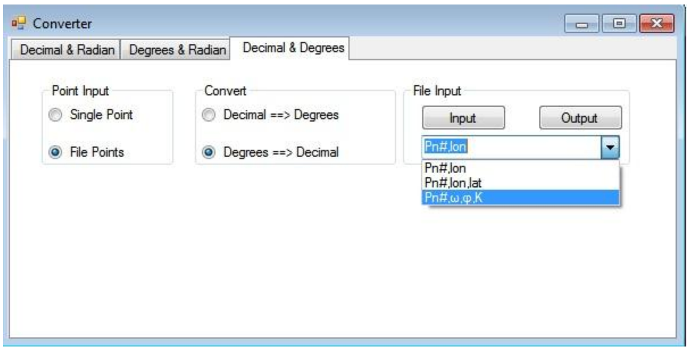

Figure (5.102): Converter dialog

6. An open dialog will show, choose the (dms to dd) file.

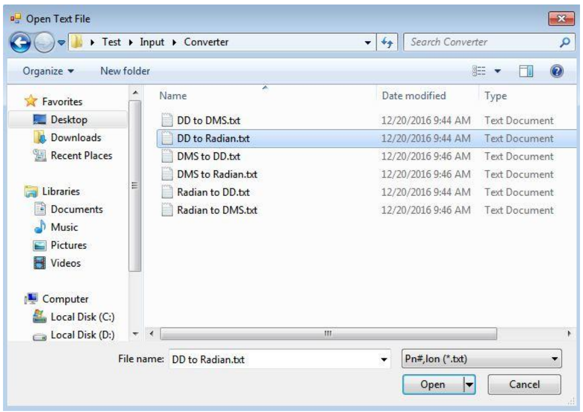

Figure (5.103): open dialog

7. Click on the Output button, and the Save as dialog will appear, write the name of the file to save in, then press save.

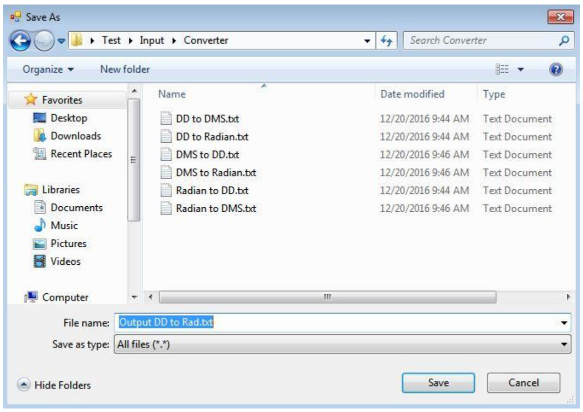

Figure (5.104): Save as dialog

8. The result will appear as shown in figure (5.105).

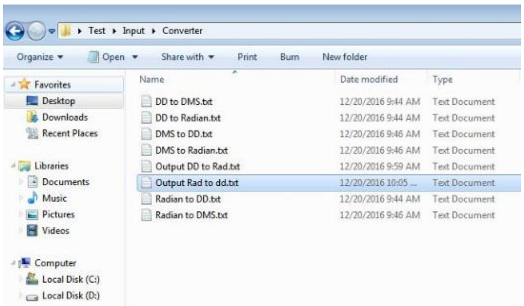

Figure (5.105): Example (5.24)

CHAPTER 6

CONCLUSIONS AND RECOMMENDATION

6.1. Conclusions

During the work & programming, the following concludes have been gain:

1. Geo-Transform V 2.0 has worked well.
2. Geo-Transform V2.0 were programmed all coordinates systems used in Palestine, and leaving the choice for a user to define his own coordinates system if not previously defined in the software.
3. Most common ellipsoids and coordinates are directly to be added in the software, to fulfill the requirements of different systems.
4. Different datum transformation solutions are used to make it more flexible to be used simply by different users.
5. Data analysis and data snooping process were successfully applied to detect blunders and analyze the networks of reference points.

6.2. Recommendations

1. In Palestine, it's recommend to use Moldonskey Transformations, because it achieves 3D transformation between GNSS with accurate ellipsoidal height (h) and Classical (old) Networks where mostly (h) is missing.
2. The Geo-Transform V 2.0 can be extended in further work to read files from different software like ArcGIS,TGO…etc.
3. Writing the Geo-Transform V 2.0 code in Java programming language or/and ObjC language so that the software can be used as an application on the mobiles that runs Android or IOS.
4. Geo-Transform V 2.0 can be extended to apply geodetic network adjustment.

DEDICATION

To Palestine…

To my Parents….

To The Soul of Martyrs….

To my Teachers ….

To my Friend's …

To whom I Love ….

To everyone who gave me Help …

To the light in the middle of the darkness

To the one who showed me the path

To Dr. Ghadi Zakarnah …

ACKNOWLEDGMENT

I would like to thank and gratitude to Allah, who gives me the most Merciful who granted me the ability and willing to start Project.

I express my thanks to Eng. Musab Shahin, who gave me the knowledge, valuable help, encouragement, supervision and guidance.

I thank "Palestine Polytechnic University", "Department of Civil and Architectural Engineering" and wish to it more progress and successes.

Thanks for all the teachers that gave me a little of their time and answered my questions.

Finally, my deep sense and sincere thanks to my parents, brothers and sisters for their patience, and for their endless support and encouragement also for everyone who tried to help me during my work and gave me strength to complete this task.

Abstract

This project aims to build geodetic Transformation software using C# programming language. This tool should be able to transform from Geographic coordinates (λ,φ,h) to Geocentric coordinates (X,Y,Z), and from Geographic coordinates to Projected coordinates (Easting, Northing), and in the opposite direction using Cassini, Transverse Mercator (TM) and Universal Transverse Mercator (UTM). So that for surveying point there would be no need to use expensive software, and the time will not be spent to draw the point in the software format like GIS and Cad software.

The projected coordinates systems used are: Palestine_1923 grid, Palestine_1923 Belt, Palestine_1923 CS Israel grid and Israel Transverse Mercator. In addition a separate datum transformation tool is to be built to calculate the datum transformation parameters using different transformations method; similarity 7-parameters, Molodensky transformation, Helmert 2D conformal transformations, Helmert 2D Affine transformation and Helmert 3D Linearized transformation.

The converter tool converts from decimal to radian and from decimal to degrees and from degrees to radian and vice versa.

There are many software available to apply these calculations, such as ArcView, ArcGIS, Autodesk Land Desktop, ER Mapper, ERDAS, ... etc. but these software are expensive for single user.

The final product will be software can be setup at any computer to calculate different types of coordinates for single or multipoint calculations.

Table of Contents

LIST OF FIGURES

LIST OF TABELS

REFERENCES:

1) Bugayevskiy, Lev M. and John P. Snyder. 1995. MapProjections: A Reference Manual. London: Tyalor and Francis.

2) Charles D. Ghilani, Adjustment Computations Spatial Data Analysis, Fifth edition, Pennsylvania State University.

3) Coordinate Systems, Global systems, Geodetic Latitude, Longitude and Height, http://www.colorado.edu/geography/gcraft/notes/coordsys/coordsys.html, visited in April 2016.

4) Dr. GhadiZakarneh, Advanced Plane Surveying, Lecture Notes, PPU, Hebron, Palestine.

5) Dr. GhadiZakarneh, Geodesy, Lecture Notes, PPU, Hebron, Palestine.

6) GáborTimár, Map grids and datums, The Burša-Wolf type datum parameters, Chapter_4 Geodetic Datums.

7) GEO 420K Class, The Global Positioning System, UT Austin, Presentation, 3-8-2006.

8) Geocentric Coordinate Systems, https://www.artima.com/forums/flat.jsp?forum=123&thread=157443, Visited in April 2016.

9) GeoZillaGeomatics Solutions and Software, WTrans, www.geozilla.de, visited in May 2016.

10) Gregory T. French, Understanding the GPS, first edition, Geo research, United States, 1996.

11) Gunter Seeber, Satellite Geodesy, second edition, de Gruyter-Berlin.

12) Huann Fan, Theoretical Geodesy, Stockholm, August 2004.

13) Hydrometronics, Topocentric Coordinates,http://www.hydrometronics.com/ecef.html, visited in April 2016.

14) Maher Owiewi, Cartography, Lecture Notes, PPU, Hebron, Palestine.

15) Marten Hooijberg, Practical Geodesy, Springer-Verlag Berlin, Heidelberg, 1997.

16) Nathaniel Bowditch, The American Practical Navigator, Chapter_15, United States Government, 2002.

17) Peter H. Dana, Map Projections,http://www.colorado.edu/geography/gcraft/notes/mapproj/mapproj_f.html, visited in April 2016.

18) Reprojection, Reprojecting geographic features,http://www.quantdec.com/Articles/reproject, visited in April 2016.

19) Simon Kalber and Reiner Jager, WTrans, 2000-2015, pdf.

20) Transit DéménagementManutention, http://www.tdm.nc, visited in April 2016.

21) USGS, Map Projections, USGS Mapping Applications Center, 28 Dec 2000, http://egsc.usgs.gov/isb//pubs/MapProjections/projections.htm, visited in May 2016.

22) Wolfgang Torge, Geodesy, second edition, de Grutyer, Berlin, New York, 1996.