

Ellipsoidal Orthographic Projection via ECEF and Topocentric (ENU)

Noel Zinn

Hydrometronics LLC

www.hydrometronics.com

June 2011

Contents

- Acknowledgements
- Motivation
- Snyder and the Orthographic
- Derivation
 - ECEF
 - Topocentric (ENU)
 - Ellipsoidal Orthographic forward and reverse
 - Example Coordinates
 - Alternative Ellipsoidal Orthographic
 - Distortion
- References
- Appendix

Acknowledgements

International Association of Oil and Gas Producers (OGP). The ellipsoidal orthographic formulas in this presentation are those of Guidance Note 7, Part 2, of the EPSG dataset provided by OGP at <http://www.epsg.org/> .

Cliff Mugnier of LSU, who led me to Waldo Tobler.

Waldo Tobler (retired), who led me to Snyder (1979).

David Burrows of ESRI, who led me to the local vertical formulas in the Manual of Photogrammetry.

ESRI, who recognize their Local Cartesian Projection (topocentric) to be the “Orthographic projection based on a spheroid”.

However, only I am responsible for any errors herein.

Orthographic Perspective of the Moon



Motivation

The orthographic perspective is the view from space, a vertical perspective from infinity with parallel rays. Probably because it is natural (our view of the moon or a globe across the room), the orthographic is regarded as aesthetically pleasing. As a map projection the orthographic is neither conformal or equal area, but there is negligible distortion near the origin. Farther from the origin, distortion can be quantified and is manageable for many applications. The orthographic is typically presented with spherical (not spheroidal or ellipsoidal) formulas.

The ellipsoidal orthographic, presented with exact formulas in this presentation, is unique among map projections. The ellipsoidal orthographic bridges the world of traditional cartography (a 3D world projected onto a 2D plane thus inducing unavoidable distortion that must be managed) and a new paradigm of 3D visualization on computers in which the world can be presented in 3D without distortion in Earth-Centered Earth-Fixed (ECEF) or topocentric (East-North-Up) coordinates. The ellipsoidal orthographic is the 2D version of 3D topocentric coordinates.

This presentation lays out the derivation of ellipsoidal orthographic formulas with scale and convergence and shows how it is derived from latitude/longitude/height to geocentric X/Y/Z (ECEF) to topocentric U/V/W (ENU) to ellipsoidal orthographic Easting (=U) and Northing (=V). The presentation suggests the use of the ellipsoidal orthographic as a useful transition from traditional cartography to 3D visualization on computers.

Snyder and the Orthographic

Wikipedia <link 1 below> offers a concise history of the orthographic map projection with spherical equations and valuable background material not covered in this presentation. Wikipedia cites Wolfram MathWorld <link 2 below>. Both cite John P. Snyder's *Map Projections - A Working Manual* (1987). In this publication Snyder treats only the spherical orthographic as is typical of many other cartographic texts.

Elsewhere, in “Calculating Map Projections for the Ellipsoid” (*American Cartographer*, April 1979), Snyder writes, “The ellipsoidal orthographic formulas are not very involved, but the projection is only useful in showing most of a hemisphere in an aesthetically-pleasing manner. Beyond the central portion, the effect of the ellipsoid is negligible, as on other azimuthals. Therefore, it is omitted here.” Too bad, but remember “not very involved”. Sixteen years later Snyder and Lev Bugayevskiy (*Map Projections – A Reference Manual*, 1995) publish ellipsoidal orthographic formulas that are very involved and (the authors acknowledge) derived with truncated series (probably not what Snyder had in mind in 1979). See Appendix for Bugayevskiy and Snyder's (approximate) ellipsoidal orthographic.

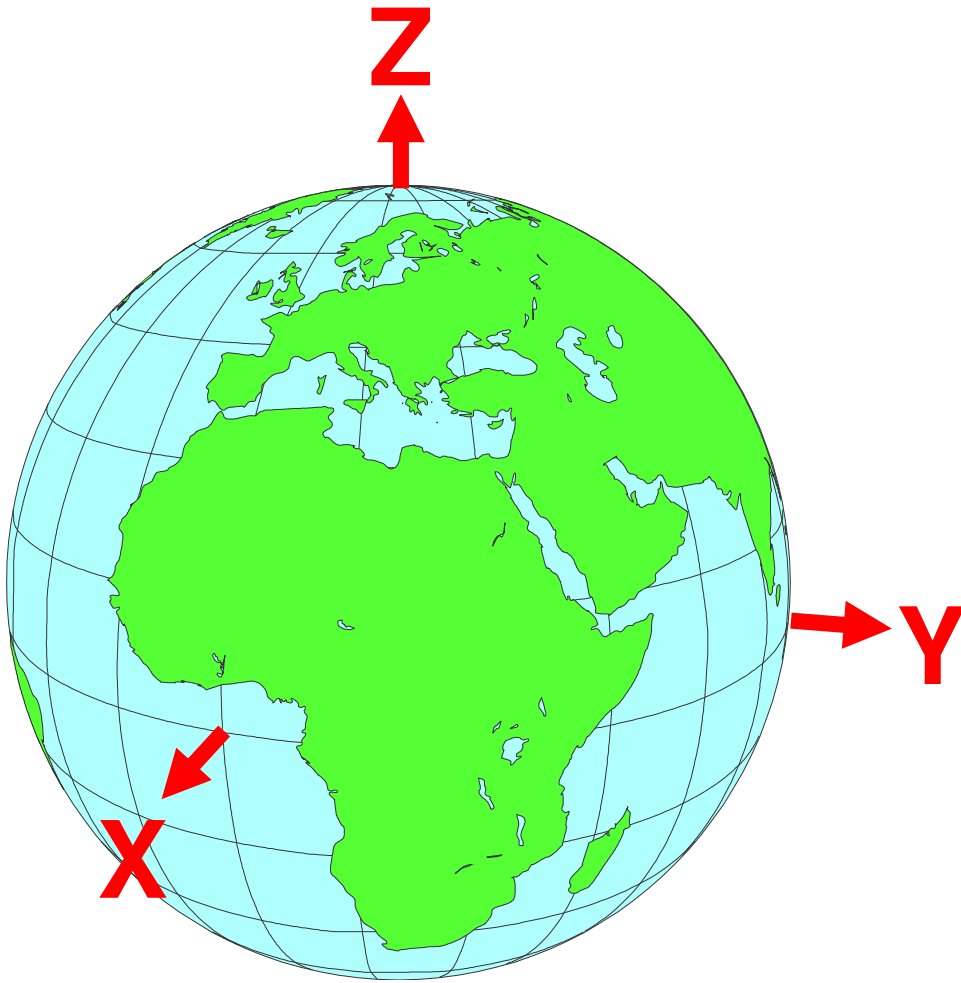
1: [http://en.wikipedia.org/wiki/Orthographic_projection_\(cartography\)](http://en.wikipedia.org/wiki/Orthographic_projection_(cartography))

2: <http://mathworld.wolfram.com/OrthographicProjection.html>

Overview of the Derivation

- Latitude (ϕ), longitude (λ) and height (L/L/H) are converted to geocentric X, Y, and Z (ECEF)
- An oblique origin for topocentric and orthographic coordinates is chosen at ϕ_0 , λ_0 , hgt=0, and the corresponding X_0 , Y_0 and Z_0 are computed
- X/Y/Z are translated and rotated from the geocenter to the oblique origin to create U/V/W (or East/North/Up), also known as topocentric coordinates
- Topocentric U is ellipsoidal orthographic Easting and topocentric V is ellipsoidal orthographic Northing. W is discarded.
- The ellipsoidal orthographic formulas can be simplified with appropriate substitutions as presented herein
- Converting U/V/W to X/Y/Z and then to L/L/H is simply a matter of reversing the computation because all information is retained
- Converting orthographic Easting and Northing to latitude and longitude is more difficult because some information is lost (viz. W)
- Therefore, this presentation takes a numerical, iterative approach to the reverse computation
- Convergence and scale are derived by differentiating the primary equations

Geocentric CRS (ECEF)



The derivation begins here. The ECEF Z-axis extends from the geocenter north along the spin axis to the North Pole. The X-axis extends from the geocenter to the intersection of the Equator and the Greenwich Meridian. The Y-axis extends from the geocenter to the intersection of the Equator and the 90E meridian.

Geographical to ECEF Coordinates

Given the ellipsoid semi-major axis (a) and flattening (f), and latitude (ϕ), longitude (λ), and height (h)

$$b = a - a \cdot f \quad e^2 = (a^2 - b^2) / a^2 \quad \nu = \frac{a}{(1 - e^2 \sin^2 \phi)^{1/2}}$$

$$X = (\nu + h) \cos \phi \cos \lambda$$

$$Y = (\nu + h) \cos \phi \sin \lambda$$

$$Z = (\nu(1 - e^2) + h) \sin \phi$$

ECEF to Geographical Coordinates

Given ellipsoid a and f , and X , Y and Z Cartesians,
a first approximation valid near the surface is:

$$b = a - a \cdot f \quad e^2 = (a^2 - b^2)/a^2 \quad e'^2 = (a^2 - b^2)/b^2$$

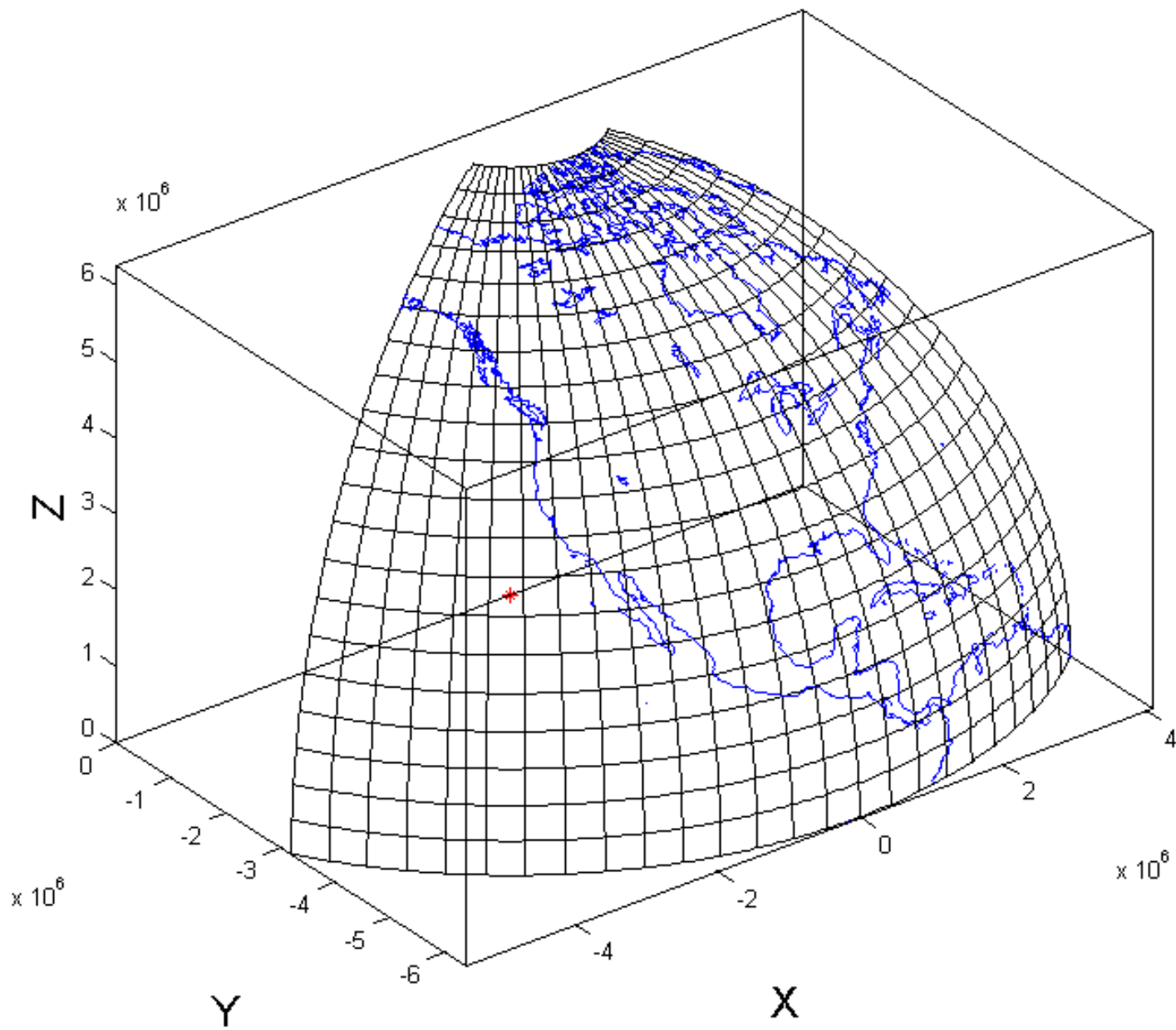
$$v = \frac{a}{(1 - e^2 \sin^2 \phi)^{1/2}} \quad p = (X^2 + Y^2)^{1/2} \quad \theta = \tan^{-1}\left(\frac{Z \cdot a}{p \cdot b}\right)$$

$$\phi = \tan^{-1} \frac{Z + e'^2 b \sin^3 \theta}{p - e^2 a \cos^3 \theta}$$

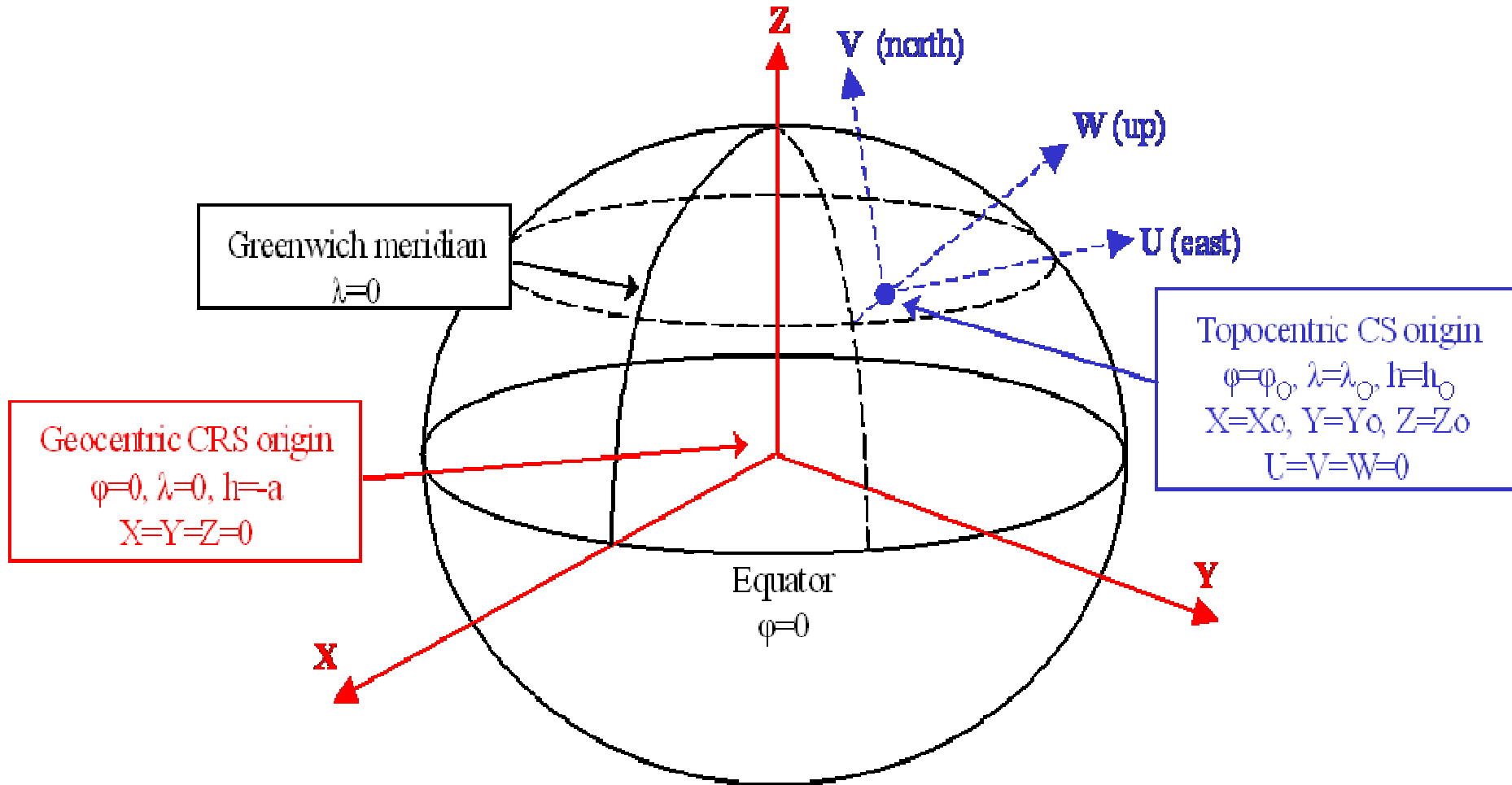
$$\lambda = \tan^{-1}\left(\frac{Y}{X}\right)$$

$$h = (p / \cos \phi) - v$$

North American ECEF Octosphere



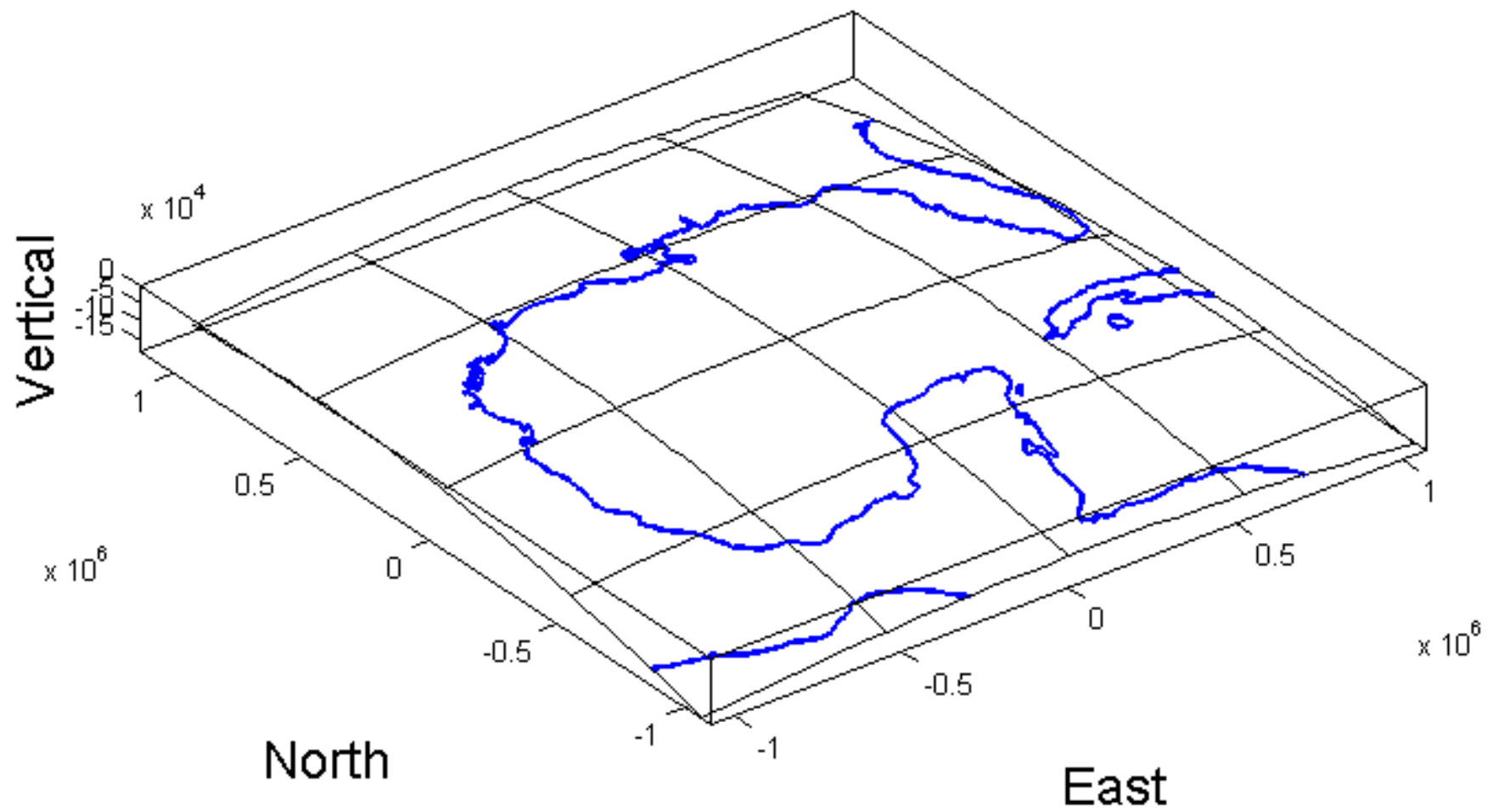
EPSG Graphic of X/Y/Z and U/V/W



Translation and Rotation to Topocentric

- There are many reasons why some users may prefer their data referenced to their local area of interest.
- Numerical precision is one reason. ECEF coordinates are large numbers that must be in double precision to maintain their resolution. ENU can be presented in single precision for a small local project.
- (There may be reasons for a large project to present in ENU, however.)
- The curvature of the Earth over a large area may be disconcerting to a user of heritage software that presents the Earth as flat (almost all geophysical software does). Zooming in to ENU coordinates presents an nearly flat world where the vertical is nearly “Up”.
- ECEF is easily translated and rotated to a topocentric reference frame. This conversion is conformal, it preserves the distortion-free curvature of the earth, and the computational burden is small, much smaller than most map projections.
- Visualization software already does something similar to change the viewing direction without recomputation of coordinates.
- The following slides present the formulas to translate and rotate from ECEF to ENU.

Gulf of Mexico in Topocentric Coordinates



Topocentric U/V/W from X/Y/Z

X/Y/Z are translated and rotated from the geocenter to the oblique origin to create U/V/W (or East/North/Up), also known as topocentric coordinates, first as a matrix expression, then the scalar equivalents

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} = \begin{bmatrix} -\sin \lambda_o & \cos \lambda_o & 0 \\ -\sin \phi_o \cos \lambda_o & -\sin \phi_o \sin \lambda_o & \cos \phi_o \\ \cos \phi_o \cos \lambda_o & \cos \phi_o \sin \lambda_o & \sin \phi_o \end{bmatrix} \cdot \begin{bmatrix} X - X_o \\ Y - Y_o \\ Z - Z_o \end{bmatrix}$$

$$U = -(X-X_o) \sin \lambda_o + (Y-Y_o) \cos \lambda_o$$

$$V = -(X-X_o) \sin \phi_o \cos \lambda_o - (Y-Y_o) \sin \phi_o \sin \lambda_o + (Z-Z_o) \cos \phi_o$$

$$W = (X-X_o) \cos \phi_o \cos \lambda_o + (Y-Y_o) \cos \phi_o \sin \lambda_o + (Z-Z_o) \sin \phi_o$$

Ellipsoidal Orthographic

In the scalar equations for U and V in the previous slide substitute X_0 , X , Y_0 , Y , Z_0 and Z for their equivalents from the equations below:

$$X = (v + h) \cos \phi \cos \lambda$$

$$Y = (v + h) \cos \phi \sin \lambda$$

$$Z = (v(1 - e^2) + h) \sin \phi$$

Reduce the result to the simplest form with appropriate substitutions (including $h = 0$ so that the plane is tangent to the ellipsoid) and get:

$$U = v \cos \phi \sin (\lambda - \lambda_0)$$

$$V = v [\sin \phi \cos \phi_0 - \cos \phi \sin \phi_0 \cos (\lambda - \lambda_0)] + e^2 (v_0 \sin \phi_0 - v \sin \phi) \cos \phi_0$$

See next slide for a full description of the ellipsoidal orthographic forward equations

Orthographic Forward

$$E = FE + v \cos \varphi \sin (\lambda - \lambda_0)$$

$$N = FN + v [\sin \varphi \cos \varphi_0 - \cos \varphi \sin \varphi_0 \cos (\lambda - \lambda_0)] + e^2 (v_0 \sin \varphi_0 - v \sin \varphi) \cos \varphi_0$$

where,

E is Easting, FE is False Easting

N is Northing, FN is False Northing

v is the prime vertical radius of curvature at latitude φ ; $v = a / (1 - e^2 \sin^2 \varphi)^{0.5}$,

v_0 is the prime vertical radius of curvature at φ_0 , $v_0 = a / (1 - e^2 \sin^2 \varphi_0)^{0.5}$,

e is the eccentricity of the ellipsoid and $e^2 = (a^2 - b^2)/a^2 = 2f - f^2$

a and b are the ellipsoidal semi-major and semi-minor axes,

$1/f$ is the inverse flattening, and

the latitude and longitude of the projection origin are φ_0 and λ_0 .

The reverse formulas are numerical and iterative.

Orthographic Reverse - 1

Seed the iteration with the center of projection (or some better guess):

$$\varphi = \varphi_0$$

$$\lambda = \lambda_0$$

Enter the iteration here with the (next) best estimates of φ and λ . Then solve for the radii of curvature in the prime vertical (v) and meridian (ρ):

$$v = a / (1 - e^2 \sin^2 \varphi)^{0.5}$$

$$\rho = a (1 - e^2) / (1 - e^2 \sin^2 \varphi)^{1.5}$$

Compute test values of E and N (E' and N') using the forward equations:

$$E' = FE + v \cos \varphi \sin (\lambda - \lambda_0)$$

$$N' = FN + v [\sin \varphi \cos \varphi_0 - \cos \varphi \sin \varphi_0 \cos (\lambda - \lambda_0)] + e^2 (v_0 \sin \varphi_0 - v \sin \varphi) \cos \varphi_0$$

Partially differentiate the forward equations to solve for the elements of the Jacobian matrix:

$$J_{11} = \partial E / \partial \varphi = -\rho \sin \varphi \sin (\lambda - \lambda_0)$$

$$J_{12} = \partial E / \partial \lambda = v \cos \varphi \cos (\lambda - \lambda_0)$$

(Continued next page ...)

Orthographic Reverse - 2

$$J_{21} = \partial N / \partial \varphi = \rho (\cos \varphi \cos \varphi_0 + \sin \varphi \sin \varphi_0 \cos (\lambda - \lambda_0))$$

$$J_{22} = \partial N / \partial \lambda = \nu \sin \varphi_0 \cos \varphi \sin (\lambda - \lambda_0)$$

Solve for the determinant of the Jacobian:

$$D = J_{11} J_{22} - J_{12} J_{21}$$

Solve the northerly and easterly differences this iteration:

$$\Delta E = E - E'$$

$$\Delta N = N - N'$$

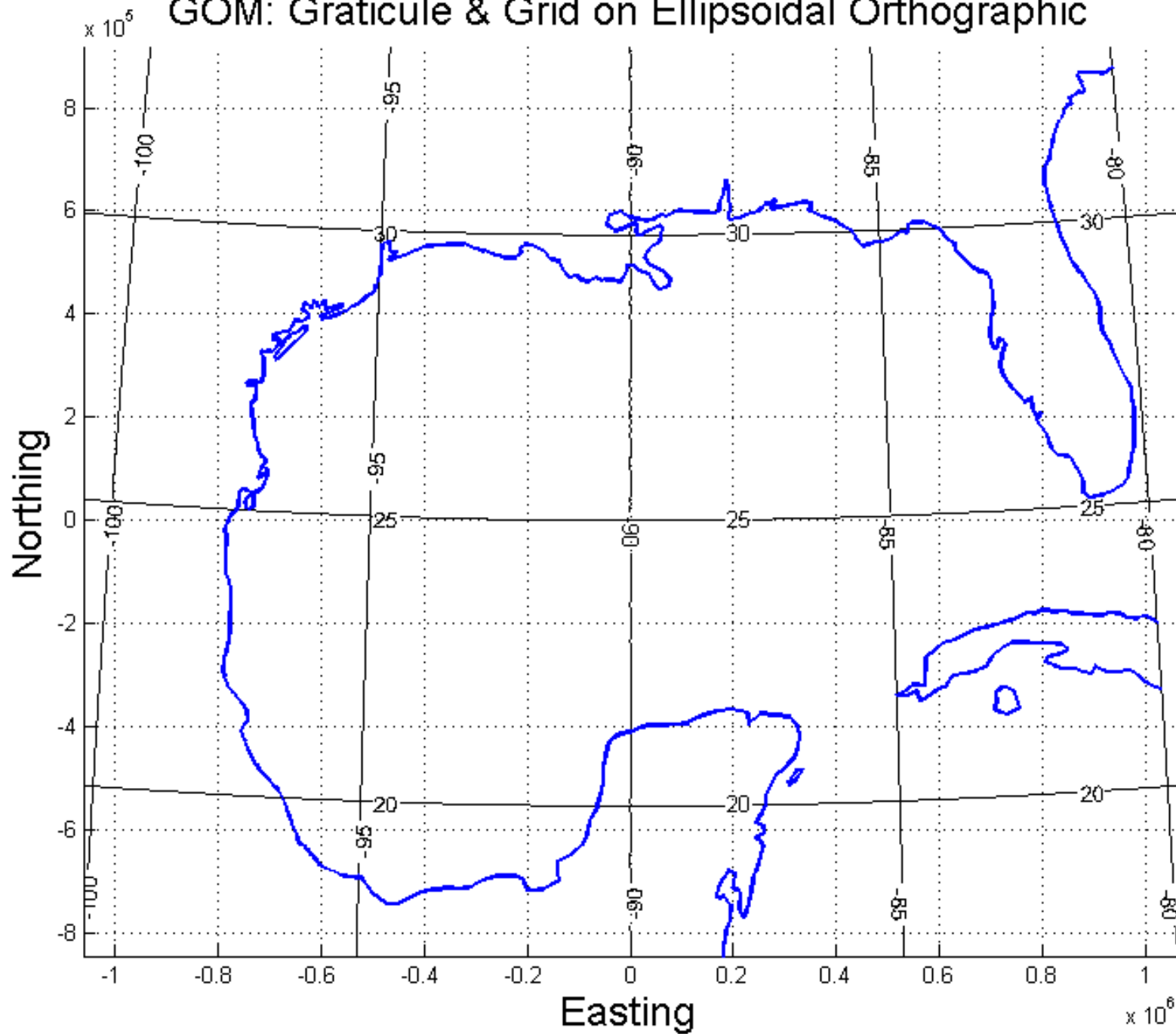
Adjust the latitude and longitude for the next iteration by inverting the Jacobian and multiplying by the differences:

$$\varphi = \varphi + (J_{22} \Delta E - J_{12} \Delta N) / D$$

$$\lambda = \lambda + (-J_{21} \Delta E + J_{11} \Delta N) / D$$

Return to the entry point with new estimates of latitude and longitude and iterate until the change in φ and λ is not significant.

GOM: Graticule & Grid on Ellipsoidal Orthographic



Example Coordinates in ECEF, Topocentric, Orthographic

3D Geocentric CRS (ECEF)

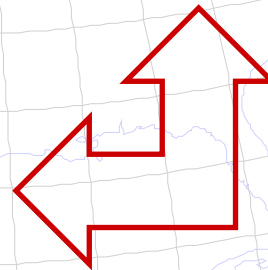
X	Y	Z
-17467.98	-5504160.95	3211700.58
-38682.38	-5507212.82	3206315.19
-46210.99	-5517257.52	3189016.48
-31331.92	-5523762.41	3177991.87
-13227.85	-5522270.08	3180692.94

3D Topocentric (ENU)

U-East	V-North	W-Up
-17467.98	600994.26	-28535.58
-38682.38	594823.66	-28045.61
-46210.99	574900.63	-26252.77
-31331.92	562159.85	-25016.54
-13227.85	565238.54	-25227.53

2D Orthographic

Easting	Northing
-17467.98	600994.26
-38682.38	594823.66
-46210.99	574900.63
-31331.92	562159.85
-13227.85	565238.54



A couple things to note. The topocentric/orthographic center is 25N/90W. 90W is the negative Y axis of ECEF. Therefore, $X = U = \text{Northing!}$ This is a special case. Note also that orthographic = topocentric without W-Up.

Orthographic Forward in Matlab - 1

```
function [xgrid, ygrid, h, k, aaxis, baxis, omega2, convg] = ...
    OrthoDirXX(A, E2, lat0rad, lon0rad, FalseN, FalseE, latrad, lonrad)

% A is semi-major axis of the ellipsoid, E2 is eccentricity squared
% lat0rad, lon0rad define the origin (radians), latrad, lonrad point to be converted (radians)
% xgrid, ygrid are Easting and Northing, FalseE, FalseN are the false coordinates
% h, k are scale factors in meridian, parallel and aaxis, baxis are min, max orthogonal scale factors
% omega2 is max angular distortion, convg is convergence

% Useful constants
d2r = pi/180; % degrees to radians
sin_lat0rad = sin(lat0rad);
cos_lat0rad = cos(lat0rad);
nu0 = A/sqrt(1-E2*sin_lat0rad^2);
const1 = nu0*E2*cos_lat0rad*sin_lat0rad;

% Pre-computations
sin_latrad = sin(latrad);
cos_latrad = cos(latrad);
sin_lonrad_minus_lon0rad = sin(lonrad-lon0rad);
cos_lonrad_minus_lon0rad = cos(lonrad-lon0rad);

% Radii
nu = A / sqrt(1 - E2*sin_latrad^2); % Prime vertical
rho = A * (1 - E2) / (sqrt(1 - E2 * sin_latrad ^ 2) ^ 3); % Meridian

% Forward equations
xgrid = FalseE + nu*cos_latrad*sin_lonrad_minus_lon0rad;
ygrid = FalseN - nu*sin_lat0rad*cos_latrad*cos_lonrad_minus_lon0rad + const1 + nu*(1-E2)*sin_latrad*cos_lat0rad;
```

Orthographic Forward in Matlab - 2

```
% Solve the partials
Xlat = -rho*sin_latrad*sin_lonrad_minus_lon0rad; % dX/dlat
Xlon = nu*cos_latrad*cos_lonrad_minus_lon0rad; % dX/dlon
Ylat = rho*(cos_latrad*cos_lat0rad + sin_latrad*sin_lat0rad*cos_lonrad_minus_lon0rad); % dY/dlat
Ylon = nu*sin_lat0rad*cos_latrad*sin_lonrad_minus_lon0rad; % dY/dlon

% Scale factor in meridian
h = sqrt(Xlat^2 + Ylat^2)*(1-E2*sin_latrad^2)^(3/2)/(A*(1-E2));
% Scale factor in parallel
k = sqrt(Xlon^2 + Ylon^2)*(1-E2*sin_latrad^2)^(1/2)/(A*cos_latrad);

% Compute the convergence in the meridians (NOT 90 degrees wrt parallel)
if Xlat == 0
    convg = 0;
else
    convg = acot(Ylat/Xlat)/d2r;
end

% Compute the intersection angle (thetaprime) of the meridians and parallels
convg1 = atan2(Ylat,Xlat);
convg2 = atan2(Ylon,Xlon);
thetaprime = convg1-convg2;

% Compute the maximum angular distortion (omega2)
aprime = sqrt(h^2+k^2+2*h*k*sin(thetaprime));
bprime = sqrt(h^2+k^2-2*h*k*sin(thetaprime));
omega2 = 2*asin(bprime/aprime)/d2r; % Using aprime and bprime

% Min and max orthogonal scale factors, NOT in parallels and meridians
aaxis = (aprime+bprime)/2;
baxis = (aprime-bprime)/2;
```

Orthographic Reverse in Matlab - 1

```
function [latrad, lonrad, h, k, aaxis, baxis, omega2, convg, testnum] = ...
    OrthoInvXX(A, E2, lat0rad, lon0rad, FalseN, FalseE, xgrid, ygrid);

% Useful constants
sin_lat0rad = sin(lat0rad);
cos_lat0rad = cos(lat0rad);
nu0 = A/sqrt(1-E2*sin_lat0rad^2);
const1 = nu0*E2*cos_lat0rad*sin_lat0rad;

% Seed with center of projection
latrad = lat0rad;
lonrad = lon0rad;

% Start the iteration
testnum = 1;
while testnum > .00001

    % Pre-computations
    sin_latrad = sin(latrad);
    cos_latrad = cos(latrad);
    sin_lonrad_minus_lon0rad = sin(lonrad-lon0rad);
    cos_lonrad_minus_lon0rad = cos(lonrad-lon0rad);

    % Radii
    nu = A / sqrt(1 - E2*sin_latrad^2);
    rho = A * (1 - E2) / (sqrt(1 - E2 * sin_latrad ^ 2) ^ 3);

    % Continued ...
```


Orthographic Reverse in Matlab - 2

```
% Test value using forward equations
xtest = FalseE + nu*cos_latrad*sin_lonrad_minus_lon0rad;
ytest = FalseN - nu*sin_lat0rad*cos_latrad*cos_lonrad_minus_lon0rad + const1 + nu*(1-E2)*sin_latrad*cos_lat0rad;

% Solve the partials
Xlat = -rho*sin_latrad*sin_lonrad_minus_lon0rad; % dX/dlat
Xlon = nu*cos_latrad*cos_lonrad_minus_lon0rad; % dX/dlon
Ylat = rho*(cos_latrad*cos_lat0rad + sin_latrad*sin_lat0rad*cos_lonrad_minus_lon0rad); % dY/dlat
Ylon = nu*sin_lat0rad*cos_latrad*sin_lonrad_minus_lon0rad; % dY/dlon

% Determinant of the Jacobian
deter = Xlat*Ylon-Xlon*Ylat;

% X/Y error this iteration
ytestnum = ygrid - ytest;
xtestnum = xgrid - xtest;
testnum = sqrt(ytestnum^2+xtestnum^2); % Radial error

% Adjust the geographicals
latrad = latrad + (Ylon*xtestnum - Xlon*ytestnum)/deter;
lonrad = lonrad + (-Ylat*xtestnum + Xlat*ytestnum)/deter;
end

% Distortion metrics computed as in the forward
```

Alternative Ellipsoidal Orthographic

All the properties of a map projection on a sphere cannot be preserved on an ellipsoid. For example, Schreiber (1866, 1897) and Krüger (1912) differed on the implementation of the conformal Transverse Mercator, varying or fixing the scale on the central meridian respectively. Snyder's conformal, ellipsoidal Stereographic differs from the conformal, ellipsoidal "Double" Stereographic used in Europe. The ellipsoidal Gnomonic can be constructed perspective or to (nearly) preserve its major property, straight great circles.

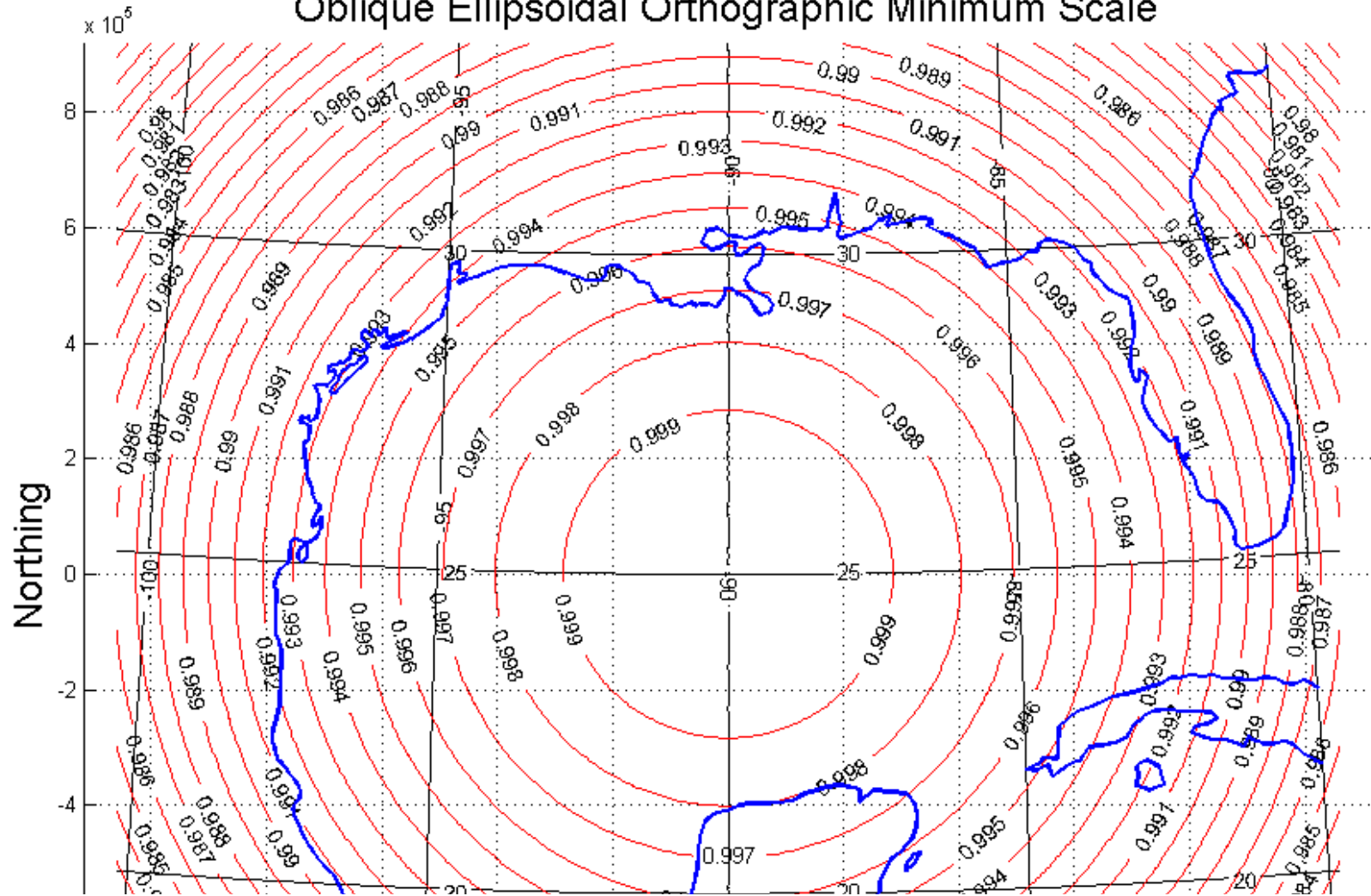
In the ellipsoidal orthographic presented here the coordinate plane is tangent to the ellipsoid. This implies that the parallel perspective rays are perpendicular to the ellipsoid at the point of tangency. It could be otherwise. Perspective rays and the normal to the ellipsoid are forever straight lines. Because ellipsoidal equipotential surfaces are not parallel, the vertical direction of gravity is a curved line. It eventually curves into the geocenter. Except at a pole or on the Equator, the normal misses the geocenter.

An ellipsoidal orthographic can be constructed with the coordinate plane secant to the ellipsoid and perpendicular to geocentric latitude, which passes through the geocenter. This alternative surrenders the valuable connection with topocentric and ECEF coordinates and is not recommended. In Bugayevskiy and Snyder – as with the EPSG – the coordinate plane is tangent to the ellipsoid.

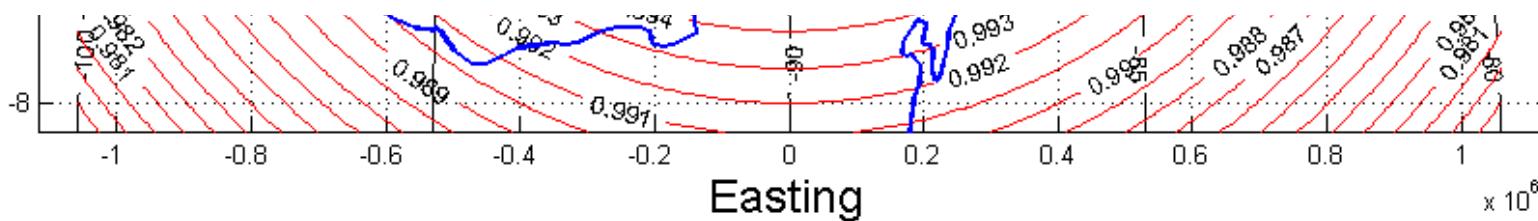
Distortions in the Following Plots

- Scale on the orthographic is everywhere unity (1) perpendicular to the radial direction, which is the direction from a point to the center of the projection. Scale decreases in the radial direction from unity at the center to zero at the horizon of the globe.
- Areal scale is the product of the two, orthogonal, linear scales and is, therefore, the same as radial scale. This plot gives the reciprocal of areal scale, which is the relative size of a “bin” on the ellipsoid with respect to that bin on the grid.
- Meridian convergence is the difference between grid north and true north at a point. Meridians and parallels are not perpendicular.
- Maximum angular distortion is the maximum difference from 90 degrees of any two, true, orthogonal directions
- Separation between the tangential plane and the ellipsoid is not a map distortion *per se*, but it is given here for comparison with topocentric coordinates. It is the “Up” or “W” dimension that is discarded in the transition between topocentric and ellipsoidal orthographic coordinates.

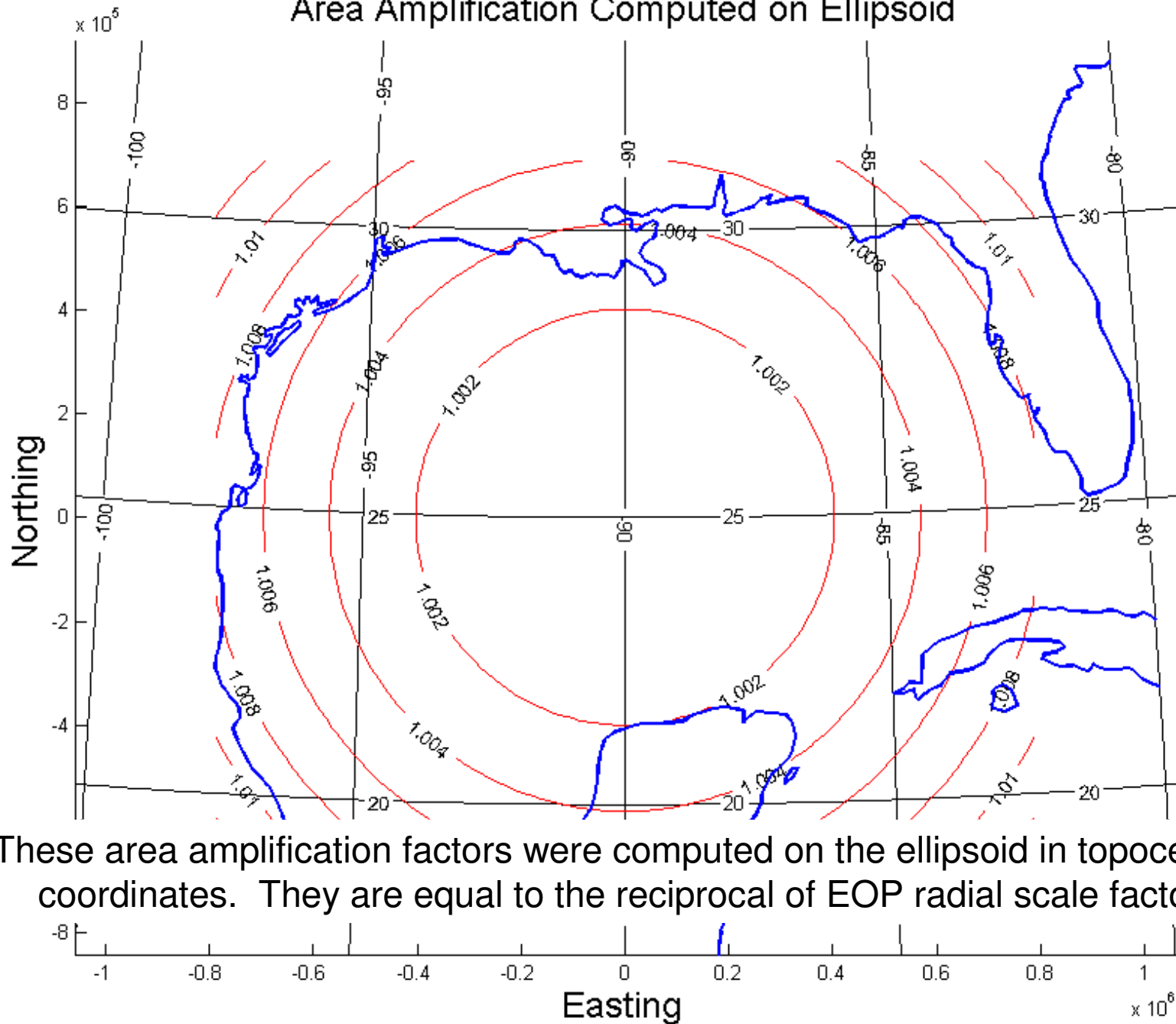
Oblique Ellipsoidal Orthographic Minimum Scale



This is scale in the radial direction. Scale in the circular direction is 1.0000

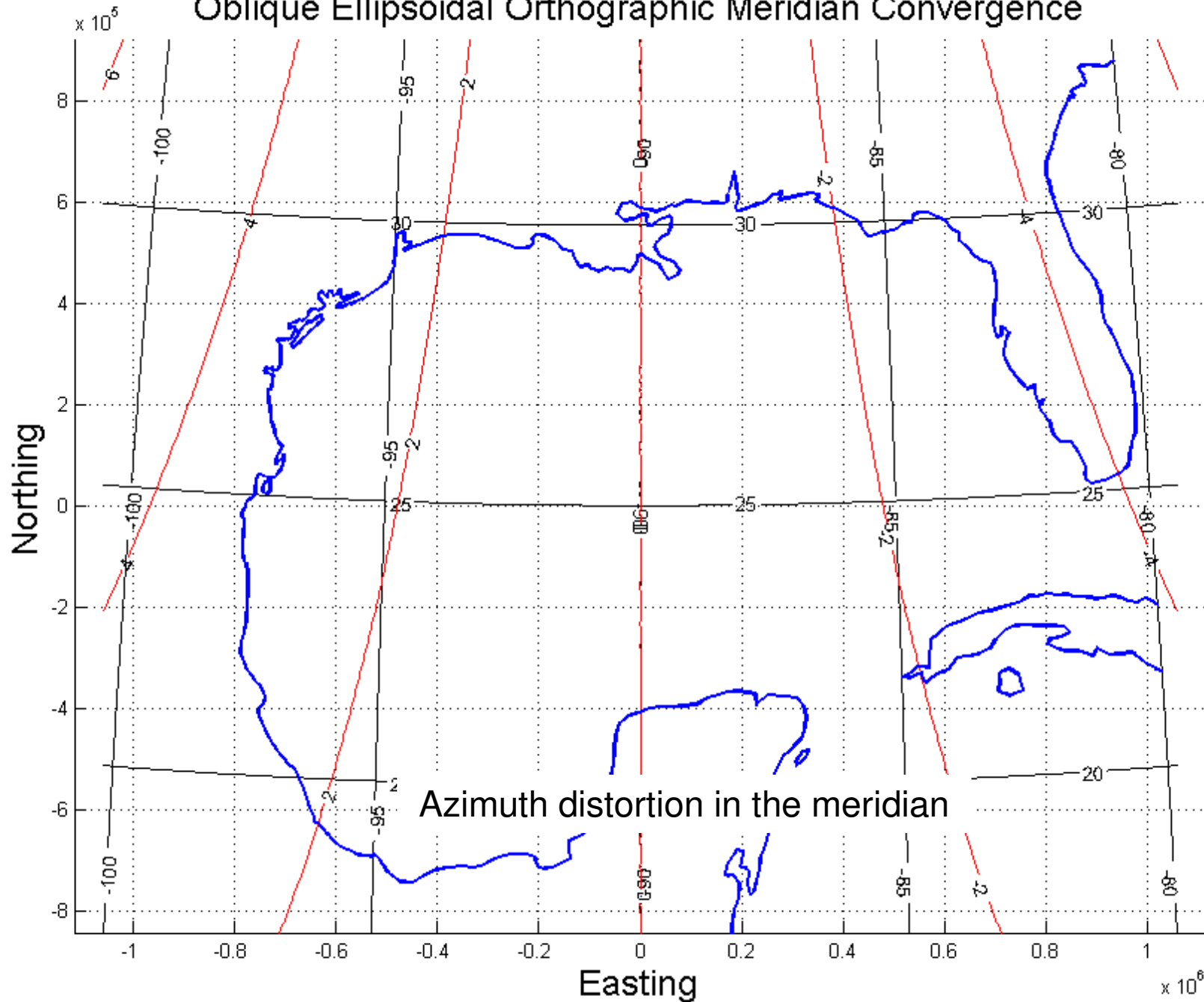


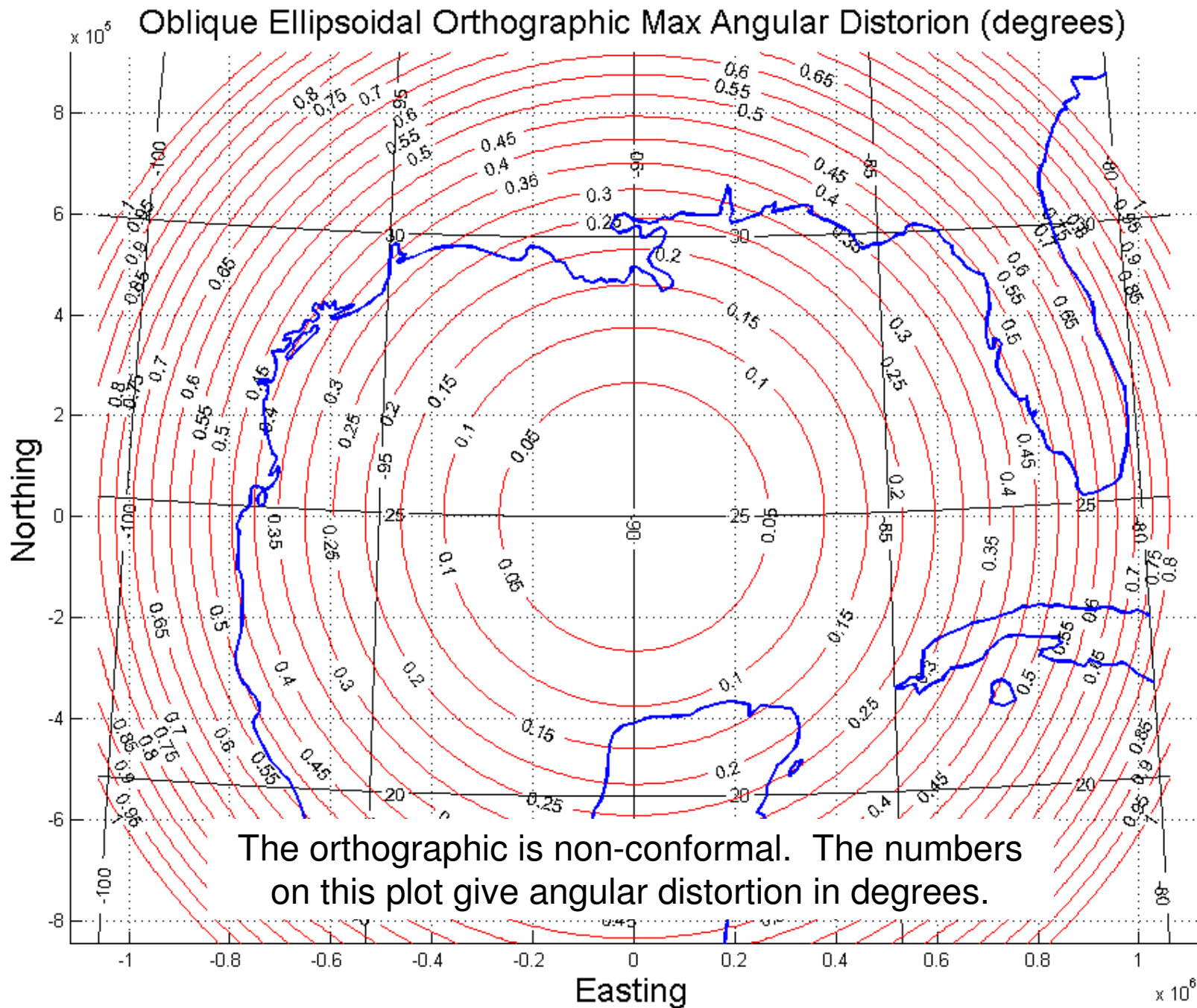
Area Amplification Computed on Ellipsoid



These area amplification factors were computed on the ellipsoid in topocentric coordinates. They are equal to the reciprocal of EOP radial scale factor.

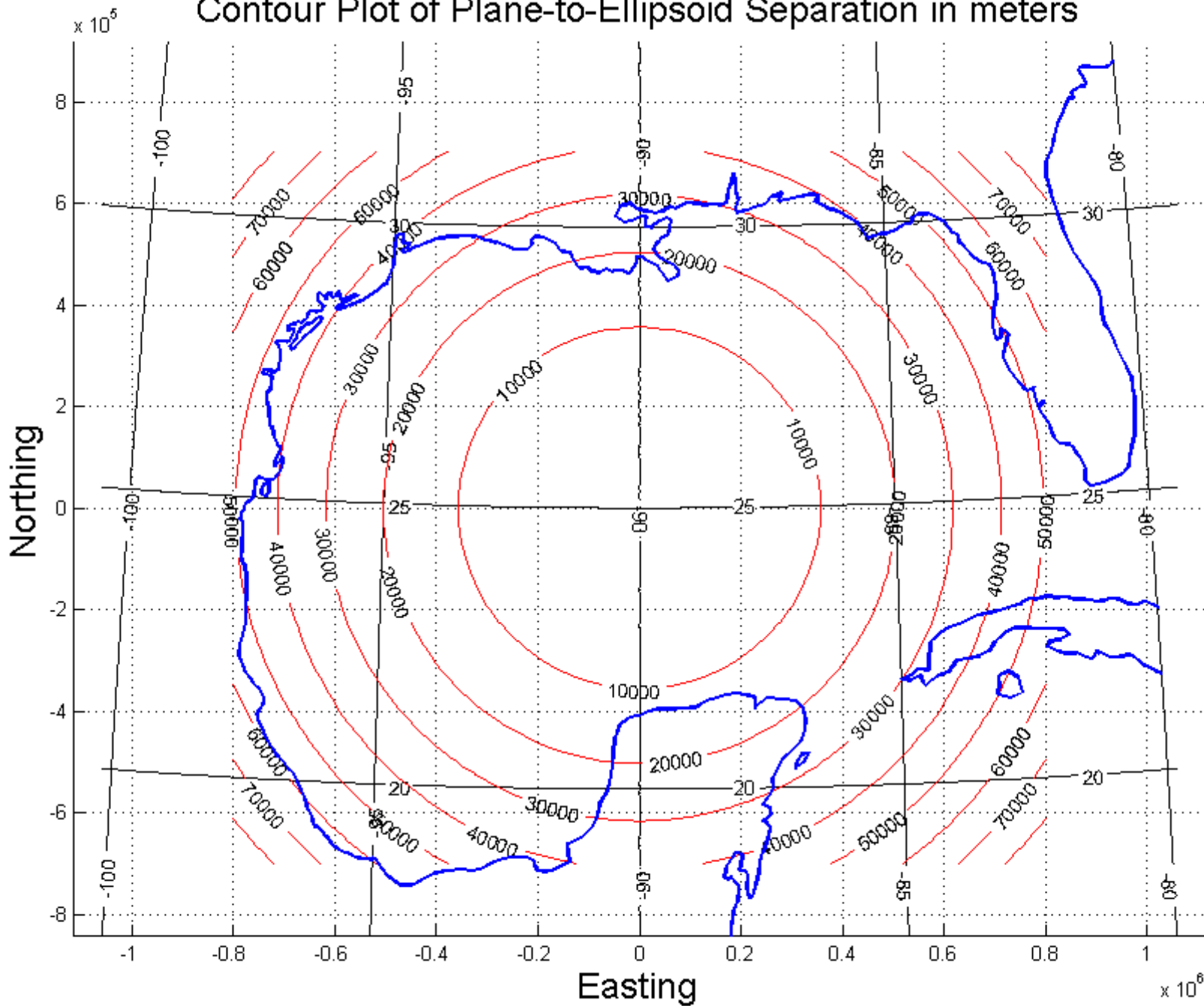
Oblique Ellipsoidal Orthographic Meridian Convergence





The orthographic is non-conformal. The numbers on this plot give angular distortion in degrees.

Contour Plot of Plane-to-Ellipsoid Separation in meters



Concluding Comments

Snyder writes of the ellipsoidal orthographic, “Beyond the central portion, the effect of the ellipsoid is negligible”. ESRI suggest that the projection is “designed for very large-scale mapping applications”, presumably near the “central portion”, and beyond a degree “distortions will greatly increase” (see Appendix). This is the conventional wisdom from the perspective precise surveying computed on conformal planes. On the other hand, a different case can be made for the projection, which deserves a second look.

First, many applications (for example, marine seismic navigation) are just not computed to the standards of precise surveying on a conformal plane. Up to $\pm 280\text{km}$ (2.5°) from the center of an orthographic, scale distortion is at worst 1m/1km, and it's 1m/4km within $\pm 140\text{km}$ of the center. That is arguably acceptable distortion over a very large area.

Second, computers, GIS and the web are changing our standards. For example, the ubiquitous, non-conformal Web Mercator, which has jumped species from the web to GIS, exhibits more angular distortion at the Equator than an orthographic does over the whole of the Gulf of Mexico. Non-conformality has become common and accepted.

Third, the orthographic is just one replaceable dimension away from topocentric and then just a rotation and translation away from ECEF, which exhibits no cartographic distortion at all. In an age of 3D visualization on computers, this is where we need to be (ECEF or ENU). The ellipsoidal orthographic projection is a transitional step in that direction.

References

- Bugayevskiy & Snyder, Map Projections – A Reference Manual, 1995
- Deakin et al, The Gauss-Kruger Projection,
<http://user.gs.rmit.edu.au/rod/files/publications/Gauss-Krueger%20Warrnambool%20Conference.pdf>
- EPSG Gudiance Note 7, Part 2, <http://www.epsg.org>
- Gerdan et al, Transforming Cartesian coordinates X,Y,Z to Geographical coordinates φ , λ , h ,
<http://user.gs.rmit.edu.au/rod/files/publications/Transforming%20Cartesian%20Coordinates.pdf>
- Karney, GeographicLib,
<http://sourceforge.net/projects/geographiclib/files/distrib/>
- Snyder, “Calculating Map Projections for the Ellipsoid”, *American Cartographer*, April 1979
- Snyder, Map Projections – A Working Manual, 1987
- Web Mercator: Non-Conformal, Non-Mercator,
[http://www.hydrometronics.com/downloads/Web%20Mercator%20-%20Non-Conformal,%20Non-Mercator%20\(notes\).pdf](http://www.hydrometronics.com/downloads/Web%20Mercator%20-%20Non-Conformal,%20Non-Mercator%20(notes).pdf)

Appendix

- ESRI Local Cartesian Projection
- Manual of Photogrammetry Local Vertical
- Wikipedia ENU
- Bugayevskiy & Snyder Topocentric Horizon and Ellipsoidal Orthographic
 - Formulas programmed in Matlab
 - Comparison with exact ellipsoidal orthographic

ESRI's Local Cartesian Projection

LOCAL CARTESIAN PROJECTION

DESCRIPTION

This is a specialized map projection that does not take into account the curvature of the earth. It's designed for very large-scale mapping applications. This projection is actually the Orthographic projection based on a spheroid.

PROJECTION METHOD

The coordinates of the center of the area of interest define the origin of the local coordinate system. The plane is tangent to the spheroid at that point, and the differences in z values are negligible between corresponding points on the spheroid and the plane. Because the differences in z values are ignored, distortions will greatly increase beyond roughly one degree from the origin.

USES AND APPLICATIONS

Large-scale mapping. In ArcInfo Workstation it should not be used for areas greater than one degree from the origin.

Manual of Photogrammetry

9.4.2.3.6.2 Geocentric-Local Vertical. As pointed out earlier, the local vertical system has an advantage (in the computation phase) over the geocentric system in that the coordinates in question have been converted to a more manageable size. In other words, a local system is no more than the geocentric system translated and rotated to some local position. In general the transformation is given by

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_L = \mathbf{M} \left\{ \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_G - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_O \right\} \quad (9.42)$$

where the vectors L , G and O represent the local system, the geocentric system and the translation. The matrix \mathbf{M} is the rotation matrix and is

$$\mathbf{M} = \begin{bmatrix} -\sin \lambda_0 & \cos \lambda_0 & 0 \\ -\sin \phi_0 \cos \lambda_0 & -\sin \phi_0 \sin \lambda_0 & \cos \phi_0 \\ \cos \phi_0 \cos \lambda_0 & \cos \phi_0 \sin \lambda_0 & \sin \phi_0 \end{bmatrix} \quad (9.43)$$

where the subscript $_0$ represents the origin's coordinates ϕ_0, λ_0 .

This is from page 485 of the 4th edition of the manual (1980).

It can be appreciated by mere inspection that Geocentric-Local Vertical of the Manual of Photogrammetry is the same as topocentric as presented herein.

Wikipedia Topocentric (ENU)

ECEF to/from ENU Coordinates

To convert from geodetic coordinates to local ENU up coordinates is a two stage process

1. Convert geodetic coordinates to ECEF coordinates
2. Convert ECEF coordinates to local ENU coordinates

To convert from local ENU up coordinates to geodetic coordinates is a two stage process

1. Convert local ENU coordinates to ECEF coordinates
2. Convert ECEF coordinates to geodetic coordinates

From ECEF to ENU

To transform from ECEF coordinates to the local coordinates we need a local reference point, typically this might be the location of a radar. If a radar is located at $\{X_r, Y_r, Z_r\}$ and an aircraft at $\{X_p, Y_p, Z_p\}$ then the vector pointing from the radar to the aircraft in the ENU frame is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\sin \lambda_r & \cos \lambda_r & 0 \\ -\sin \phi_r \cos \lambda_r & -\sin \phi_r \sin \lambda_r & \cos \phi_r \\ \cos \phi_r \cos \lambda_r & \cos \phi_r \sin \lambda_r & \sin \phi_r \end{bmatrix} \begin{bmatrix} X_p - X_r \\ Y_p - Y_r \\ Z_p - Z_r \end{bmatrix}$$

Bugayevskiy & Snyder (B&S) Ellipsoidal Orthographic

Bugayevskiy & Snyder nail “topocentric horizon” coordinates (ENU, and thus the ellipsoidal orthographic) on page 3 of their reference manual, though in a form somewhat different than the other sources cited. Despite that, B&S provide alternative, more involved equations for the orthographic on pages 4, 5, 115 & 116 as part of a development of a family of perspectives including the gnomonic and the stereographic. It is challenging to extract the formulas essential to the ellipsoidal orthographic from this development. Those are presented as Matlab code on the next slide. (Prudence prevents me from photocopying the text.)

More than once in this development B&S note that certain series are truncated at the e^2 term (i.e. no higher terms), e.g. “When solving problems of cartography, photogrammetry, and some problems of geodesy ... it is sufficient to include terms of up to e^2 ” (page 5). Because of the authors’ sprawling development, I cannot be certain that these caveats apply to the equations essential to my Matlab code. The B&S orthographic differs from the EPSG orthographic by about 1:200,000 in Northing and 1:6,000,000 in Easting. I attribute this to approximations. A difference plot is presented.

B&S Ellipsoidal Orthographic

```
function [xgrid, ygrid] = ...
    BOrthoDir(A, E2, lat0rad, lon0rad, FalseN, FalseE, latrad, lonrad);

% A is semi-major axis of the ellipsoid, E2 is eccentricity squared
% lat0rad, lon0rad define the origin (radians)
% latrad, lonrad point to be converted (radians)
% FalseE, FalseN are the false coordinates
% xgrid, ygrid are Easting and Northing

% Compute the radius of curvature in prime vertical at lat0
N0 = A/sqrt(1-E2*sin(lat0rad)^2);
% Compute other B&S terms relevant to the orthographic
t1 = sin(latrad)*cos(lat0rad) - cos(latrad)*sin(lat0rad)*cos(lonrad-lon0rad);
t4 = cos(latrad)*sin(lonrad-lon0rad);
tau = sin(latrad) - sin(lat0rad);

% Compute grid coordinates on the ellipsoid
xgrid = FalseE + N0*( t4*( 1 + (E2/2)*tau*(2*sin(latrad) - tau)));
ygrid = FalseN + N0*( t1 + (E2/2)*tau*( 2*( t1*sin(latrad) - cos(lat0rad)) - tau*t1));
```


EPSG-B&S Orthographic: $\Delta N/\Delta E$

