Got Geodetics?

Contents

- Motivation for using an efficient geodetic coordinate transformation

- Performance Summary
  - C/C++ Implementation
  - FORTRAN Implementation
  - MATLAB Implementation
  - Python3 Implementation
  - Julia Implementation

- Summary geodetic coordinate transformation details
  - Preliminaries
  - Determine the x cartesian coordinate of point P in terms of the geodetic latitude φ
  - Determine the y cartesian coordinate of point P in terms of the geodetic latitude φ
  - Determine various lengths in terms of geodetic latitude φ
  - Relationships between various latitudes
  - Summary
  - Geodetic Coordinates { φ λ h } Converted To Cartesian Coordinates { x y z }
  - Basic Geodetic Equations
  - Geodetic Altitude h in terms of Latitudes φ and θ (alternate form)
  - Geodetic Latitude φ in terms of Geocentric Latitude θ (alternate form)
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Geodetic Longitude λ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Geodetic Altitude h Given Geodetic Latitude φ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Geodetic Equations Given Cartesian Coordinates { x y z }
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Relationships Between Geodetic Latitude φ and Reduced Latitude ψ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Transform Geodetic Latitude φ Equation
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Solution For Transformed Geodetic Latitude φ Equation
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Equations for p and z in terms of Geodetic Latitudes φ and Reduced Latitude ψ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Halley's Third Order Method : Starting Value T_0
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Solution For Transformed Geodetic Latitude φ Equation

Contents

- Summary geodetic coordinate transformation details (continued)
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Geodetic Altitude h Given Geodetic Latitude φ and Reduced Latitude ψ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Solution For Transformed Geodetic Latitude φ Equation
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Geodetic Altitude h Given Geodetic Latitude φ
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Solution For Transformed Geodetic Altitude h (continued)
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Halley's Third Order Method Algorithm Operation Count
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Halley's Third Order Method Algorithm Separated Into Numerator and Denominator
  - Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Halley's Third Order Method Algorithm Code Implementation

- Implementations of one efficient geodetic coordinate transformation.
  1. C / C++ implementation
  2. Fortran implementation
  3. Python implementation
  4. Julia implementation
  5. Matlab implementation

- References

Motivation for using an efficient geodetic coordinate transformation

Reference [1] Abstract:

By using Halley's third-order formula to find the root of a non-linear equation, we develop a new iterative procedure to solve an irrational form of the "latitude equation", the equation to determine the geodetic latitude for given Cartesian coordinates. With a limit to one iteration, starting from zero height, and minimizing the number of divisions by means of the rational form representation of Halley's formula, we obtain a new non-iterative method to transform Cartesian coordinates to geodetic ones. The new method is sufficiently precise in the sense that the maximum error of the latitude and the relative height is less than 6 micro-arcseconds for the range of height, -10 km $\leq h \leq 30{,}000$ km. The new method is around 50% faster than our previous method, roughly twice as fast as the well-known Bowring's method, and much faster than the recently developed methods of Borkowski, Laskowski, Lin and Wang, Jones, Pollard, and Vermeille.

Motivation For An Efficient Implementation   (Taken from Introduction of Reference [1]):

GPS and other space-geodetic techniques have enhanced the importance of the transformation between the geodetic coordinates (ϕ, λ, h) and Cartesian rectangular coordinates (x, y, z). For instance, some mobile phones now incorporate a GPS receiver in order to provide on-line location information to the end-users. In Japan, the Ministry of General Affairs announced in May 2004 that all third-generation mobile phones in Japan must contain a GPS receiver by April 2007. This policy is to identify the location of emergency calls from mobile phones. Since the computational resource is limited in a mobile phone and the frequency of calling the transformation is tremendously high, the development of fast, concise, and yet accurate procedures is urgently needed.

Use of Third Order Halley's Method   (Taken from Section 2 of Reference [1]):

In general, it is very time-consuming to evaluate the second derivative rigorously. This is the reason why Halley's method is not popular, although it provides a cubic rate of convergence to the solution. Fortunately $g''(T)$ is evaluated with only a small increase in the computation time. As such, Halley's method is expected to be more efficient than Newton's method in this case.

Performance Summary - C / C++ Implementation

C / C++:

```
CPU: "Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz"

 • 8.3 [nanoseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

```
CPU: "Intel(R) Xeon(R) W-2135 CPU @ 3.70GHz"

 • 0.56 [nanoseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

Performance Summary - FORTRAN Implementation

FORTRAN:

```
CPU: "Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz"

 • 0.347 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

```
CPU: "Intel(R) Xeon(R) W-2135 CPU @ 3.70GHz"

 • 0.467 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

Performance Summary - MATLAB Implementation

MATLAB:

```
CPU: "Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz"

 • 23.51 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

```
CPU: "Intel(R) Xeon(R) W-2135 CPU @ 3.70GHz"

 • 7.036 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

Performance Summary – Python3 Implemenation

Python3:

```
CPU: "Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz"

 • 126.403 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

```
CPU: "Intel(R) Xeon(R) W-2135 CPU @ 3.70GHz"

 • 69.16 [microseconds] per geocentric rectangular to geodetic conversion
 • Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
 • Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]
```

Performance Summary - Julia Implementation

Julia:

CPU: "Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz"

- 276.85 [nanoseconds] per geocentric rectangular to geodetic conversion
- Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
- Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]

CPU: "Intel(R) Xeon(R) W-2135 CPU @ 3.70GHz"

- 193.22 [nanoseconds] per geocentric rectangular to geodetic conversion
- Maximum geodetic north latitude absolute error over all trials is:--> +1.044606e+00 [microarcseconds]
- Maximum geodetic altitude absolute error over all trials is:--------> +2.793968e+00 [nanometers]

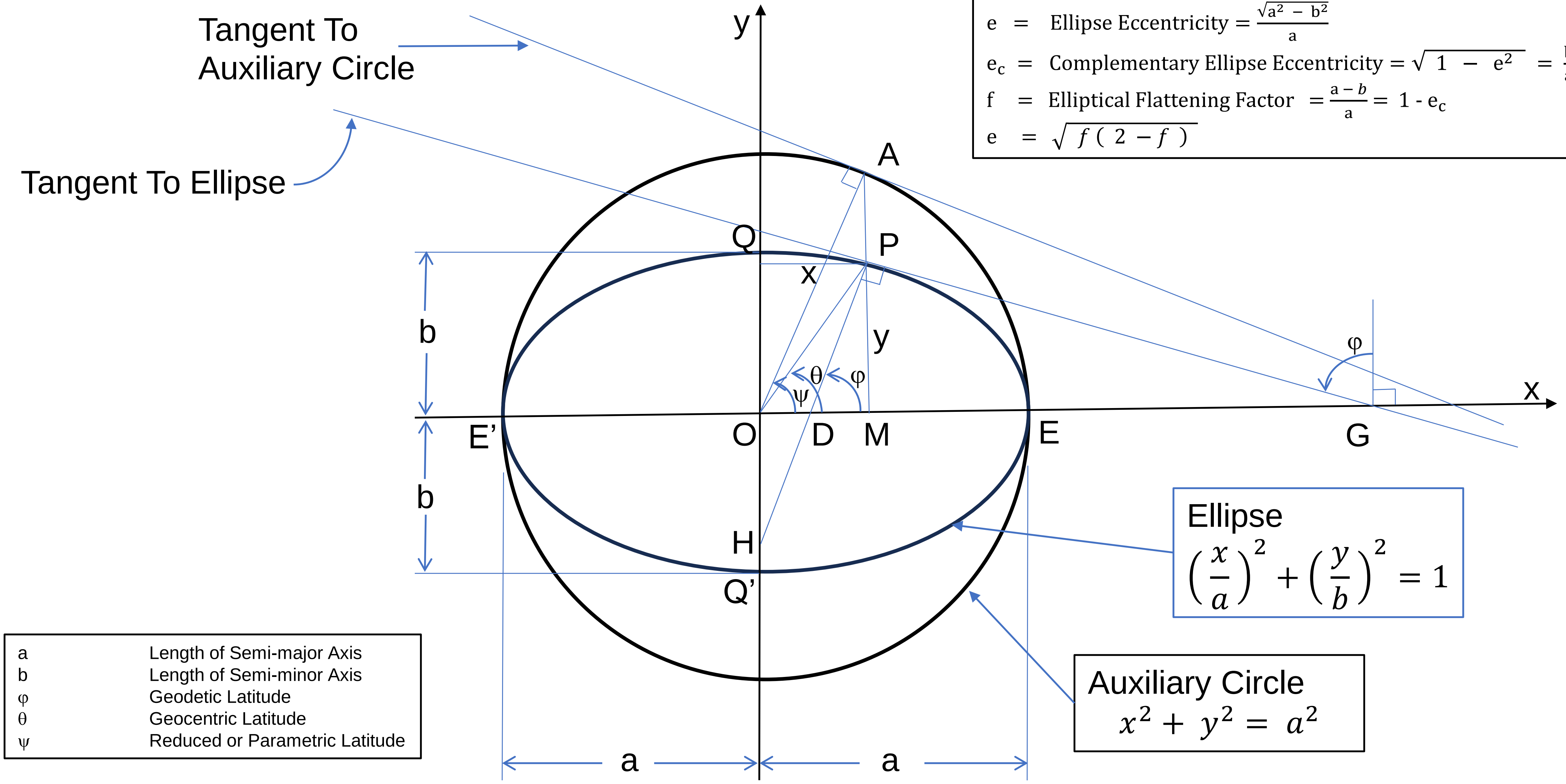
Summary geodetic coordinate transformation details - Preliminaries

e = Ellipse Eccentricity $= \dfrac{\sqrt{a^2 - b^2}}{a}$

e_c = Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \dfrac{b}{a}$

f = Elliptical Flattening Factor $= \dfrac{a - b}{a} = 1 - e_c$

$e = \sqrt{f(2 - f)}$



a	Length of Semi-major Axis
b	Length of Semi-minor Axis
φ	Geodetic Latitude
θ	Geocentric Latitude
ψ	Reduced or Parametric Latitude

Determine the x cartesian coordinate of point P in terms of the geodetic latitude φ

The equation for an ellipse is: $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$

Differentiating with respect to x: $\frac{2x}{a^2} + \frac{2y}{b^2}\frac{dy}{dx} = 0$

Re-arranging: $\frac{dy}{dx} = -\frac{b^2}{a^2}\,\frac{x}{y}$

By definition, $\frac{dy}{dx}$ is the slope of the tangent to the ellipse, thus:

$$\frac{dy}{dx} = \tan(\varphi + 90°) = -\cot(\varphi)$$

Thus: $-\frac{b^2}{a^2}\,\frac{x}{y} = -\cot(\varphi)$

$$\frac{x}{y} = \frac{a^2}{b^2}\cot(\varphi) \quad \text{or} \quad \frac{y}{x} = \frac{b^2}{a^2}\frac{1}{\cot(\varphi)} = \frac{b^2}{a^2}\tan(\varphi)$$

So: $y = \frac{b^2}{a^2}\frac{x}{\cot(\varphi)} = \frac{b^2}{a^2}\tan(\varphi)\, x$

And: $x = \frac{a^2}{b^2}\frac{y}{\tan(\varphi)}$

Substituting $y = \frac{b^2}{a^2}\tan(\varphi)\, x$ into $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ yields:

$$1 = \frac{x^2}{a^2} + \frac{\left[\frac{b^2}{a^2}\tan(\varphi)\, x\right]^2}{b^2} = \frac{x^2}{a^2} + \frac{b^2}{a^4}[\tan(\varphi)]^2 x^2 = \frac{x^2}{a^2}\left[1 + \frac{b^2}{a^2}\frac{[\sin(\varphi)]^2}{[\cos(\varphi)]^2}\right]$$

But $\frac{b^2}{a^2} = 1 - e^2$ so $1 = \frac{x^2}{a^2}\left[1 + (1 - e^2)\frac{[\sin(\varphi)]^2}{[\cos(\varphi)]^2}\right]$

And $1 = \frac{x^2}{a^2}\left[\frac{[\cos(\varphi)]^2 + [\sin(\varphi)]^2 - e^2[\sin(\varphi)]^2}{[\cos(\varphi)]^2}\right] = \frac{x^2}{a^2}\left[\frac{1 - e^2[\sin(\varphi)]^2}{[\cos(\varphi)]^2}\right]$

Finally:

$$\boxed{\mathbf{x} = \frac{a\cos(\varphi)}{\sqrt{1 - e^2[\sin(\varphi)]^2}}}$$

Determine the y cartesian coordinate of point P in terms of the geodetic latitude φ

From the previous chart: $y = \frac{b^2}{a^2}\frac{x}{\cot(\varphi)} = \frac{b^2}{a^2}\tan(\varphi)\, x$

And: $x = \frac{a^2}{b^2}\frac{y}{\tan(\varphi)}$

Substituting $x = \frac{a^2}{b^2}\frac{y}{\tan(\varphi)}$ into $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ yields:

$$1 = \frac{\left[\frac{a^2}{b^2}\frac{y}{\tan(\varphi)}\right]^2}{a^2} + \frac{y^2}{b^2} = \frac{\left[\frac{a^4}{b^4}\ \frac{y^2}{\tan^2(\varphi)}\right]}{a^2} + \frac{y^2}{b^2} = \frac{a^2}{b^4}\ \frac{y^2}{\tan^2(\varphi)} + \frac{y^2}{b^2} = \frac{y^2}{b^2}\left[1 + \frac{a^2}{b^2}\frac{[\cos(\varphi)]^2}{[\sin(\varphi)]^2}\right]$$

But $\frac{a^2}{b^2} = \frac{1}{1-e^2}$ so $1 = \frac{y^2}{b^2}\left[1 + \frac{1}{1-e^2}\frac{[\cos(\varphi)]^2}{[\sin(\varphi)]^2}\right] = \frac{y^2}{b^2}\left[1 + \frac{[\cos(\varphi)]^2}{(1-e^2)[\sin(\varphi)]^2}\right]$

And $$1 = \frac{y^2}{b^2}\left[\frac{[\sin(\varphi)]^2 - e^2[\sin(\varphi)]^2 + [\cos(\varphi)]^2}{(1-e^2)[\sin(\varphi)]^2}\right] = \frac{y^2}{b^2}\left[\frac{1 - e^2[\sin(\varphi)]^2}{(1-e^2)[\sin(\varphi)]^2}\right]$$

Finally: $$\boxed{\mathbf{y} = \frac{\mathbf{b}\ \sqrt{1-e^2}\ \sin(\varphi)}{\sqrt{1 - e^2[\sin(\varphi)]^2}}}$$

Determine various lengths in terms of geodetic latitude φ

ength of normal terminating on minor axis (PH): Length PH $= \dfrac{x}{\cos(\varphi)} = \dfrac{\dfrac{a\cos(\varphi)}{\sqrt{1-e^2[\sin(\varphi)]^2}}}{\cos(\varphi)} = \boxed{\dfrac{\mathbf{a}}{\sqrt{\mathbf{1-e^2[\sin(\varphi)]^2}}}}$

Length of normal terminating on major axis (PD): Length PD $= \dfrac{y}{\sin(\varphi)} = \dfrac{\dfrac{b\sqrt{1-e^2}\,\sin(\varphi)}{\sqrt{1-e^2[\sin(\varphi)]^2}}}{\sin(\varphi)} = \dfrac{b\sqrt{1-e^2}}{\sqrt{1-e^2[\sin(\varphi)]^2}} = \dfrac{b\sqrt{\frac{b^2}{a^2}}}{\sqrt{1-e^2[\sin(\varphi)]^2}} = \dfrac{\frac{b^2}{a}}{\sqrt{1-e^2[\sin(\varphi)]^2}}$

$$= \frac{a\,\frac{b^2}{a^2}}{\sqrt{1-e^2[\sin(\varphi)]^2}} = \frac{a\,(1-e^2)}{\sqrt{1-e^2[\sin(\varphi)]^2}}$$

$$= (1-e^2)\frac{a}{\sqrt{1-e^2[\sin(\varphi)]^2}} = \boxed{\mathbf{(1-e^2)\ Length\ PH}}$$

Length of DH = Length of PH – Length of PD = $\dfrac{a}{\sqrt{1-e^2[\sin(\varphi)]^2}} - (1-e^2)\dfrac{a}{\sqrt{1-e^2[\sin(\varphi)]^2}} = \dfrac{a}{\sqrt{1-e^2[\sin(\varphi)]^2}}\,e^2 = \boxed{\boldsymbol{e^2}\ \mathbf{Length\ PH}}$

Length of OH = (Length of DH) $\sin(\varphi) = \boxed{\dfrac{\mathbf{a}}{\sqrt{\mathbf{1-e^2[\sin(\varphi)]^2}}}\boldsymbol{e^2}\mathbf{\sin(\varphi)}} = \boxed{\boldsymbol{e^2}\mathbf{\sin(\varphi)\ \ Length\ PH}} = \boxed{\mathbf{\sin(\varphi)\ \ Length\ DH}}$

Relationships between various latitudes

Parametric equations for the ellipse in terms of the reduced or parametric latitude ψ are: $\boxed{\mathbf{x = a\cos(\psi)}}$ and $\boxed{\mathbf{y = b\sin(\psi)}}$

Differentiating these parametric equations with respect to the reduced latitude ψ : $\frac{dx}{d\psi} = -a\sin(\psi)$ and $\frac{dy}{d\psi} = b\cos(\psi)$

Then: $$\frac{dy}{dx} = \frac{dy}{d\psi}\frac{d\psi}{dx} \overset{?}{=} \frac{\frac{dy}{d\psi}}{\frac{dx}{d\psi}} = \frac{b\cos(\psi)}{-a\sin(\psi)} = -\frac{b}{a}\frac{\cos(\psi)}{\sin(\psi)} = -\frac{b}{a}\cot(\psi)$$

By definition, $\frac{dy}{dx}$ is the slope of the tangent to the ellipse and, $\frac{dy}{dx} = \tan(\varphi + 90°) = -\cot(\varphi)$

So now the reduced latitude ψ can be related to the geodetic latitude φ : $-\cot(\varphi) = \frac{dy}{dx} = -\frac{b}{a}\cot(\psi)$ so $\cot(\varphi) = \frac{b}{a}\cot(\psi)$

and

or $\frac{1}{\tan(\varphi)} = \frac{b}{a}\frac{1}{\tan(\psi)}$ so $\tan(\psi) = \frac{b}{a}\tan(\varphi)$

This gives relationships between geodetic latitude φ and reduced latitude ψ : $$\boxed{\mathbf{\tan(\psi) = \frac{b}{a}\tan(\varphi) = \sqrt{1 - e^2}\tan(\varphi) = (1 - f)\tan(\varphi)}}$$

Also: $$\tan(\theta) = \frac{y}{x} = \frac{\frac{b\sqrt{1-e^2}\sin(\varphi)}{\sqrt{1-e^2[\sin(\varphi)]^2}}}{\frac{a\cos(\varphi)}{\sqrt{1-e^2[\sin(\varphi)]^2}}} = \frac{b\sqrt{1-e^2}\sin(\varphi)}{\sqrt{1-e^2[\sin(\varphi)]^2}}\,\frac{\sqrt{1-e^2[\sin(\varphi)]^2}}{a\cos(\varphi)}$$

$$= \frac{b\sqrt{1-e^2}\sin(\varphi)}{a\cos(\varphi)} = \frac{b}{a}\sqrt{1-e^2}\tan(\varphi)$$

$$= \sqrt{1-e^2}\sqrt{1-e^2}\tan(\varphi) = (1-e^2)\tan(\varphi) = \frac{b^2}{a^2}\tan(\varphi)$$

This gives relationships between geodetic latitude φ and geocentric latitude θ: $$\boxed{\mathbf{\tan(\theta) = (1 - e^2)\tan(\varphi) = \frac{b^2}{a^2}\tan(\varphi) = (1 - f)^2\tan(\varphi)}}$$

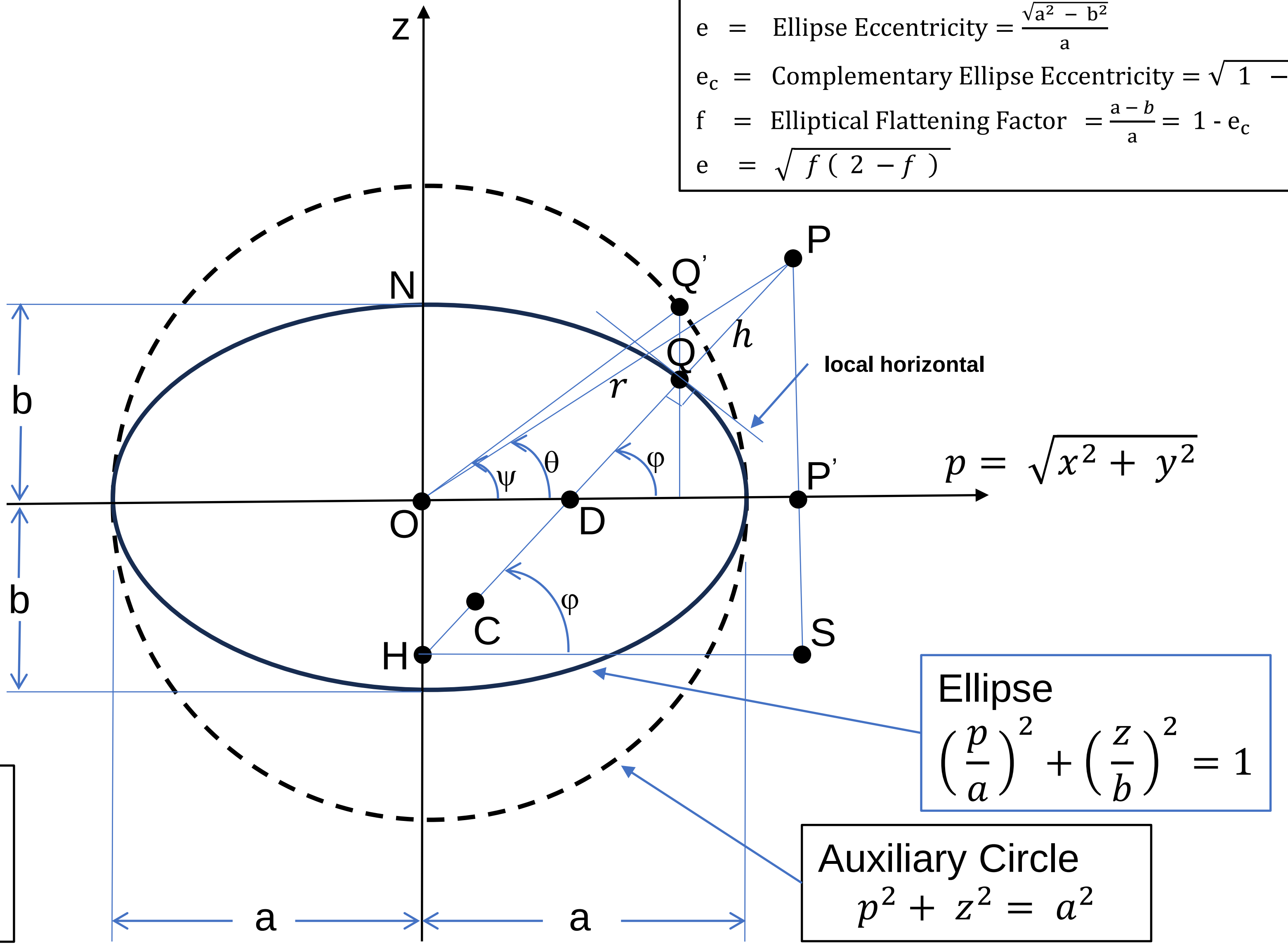
Summary

$$e \;=\; \text{Ellipse Eccentricity} = \frac{\sqrt{a^2 - b^2}}{a}$$

$$e_c \;=\; \text{Complementary Ellipse Eccentricity} = \sqrt{1 - e^2} \;=\; \frac{b}{a}$$

$$f \;=\; \text{Elliptical Flattening Factor} \;=\; \frac{a - b}{a} = 1 - e_c$$

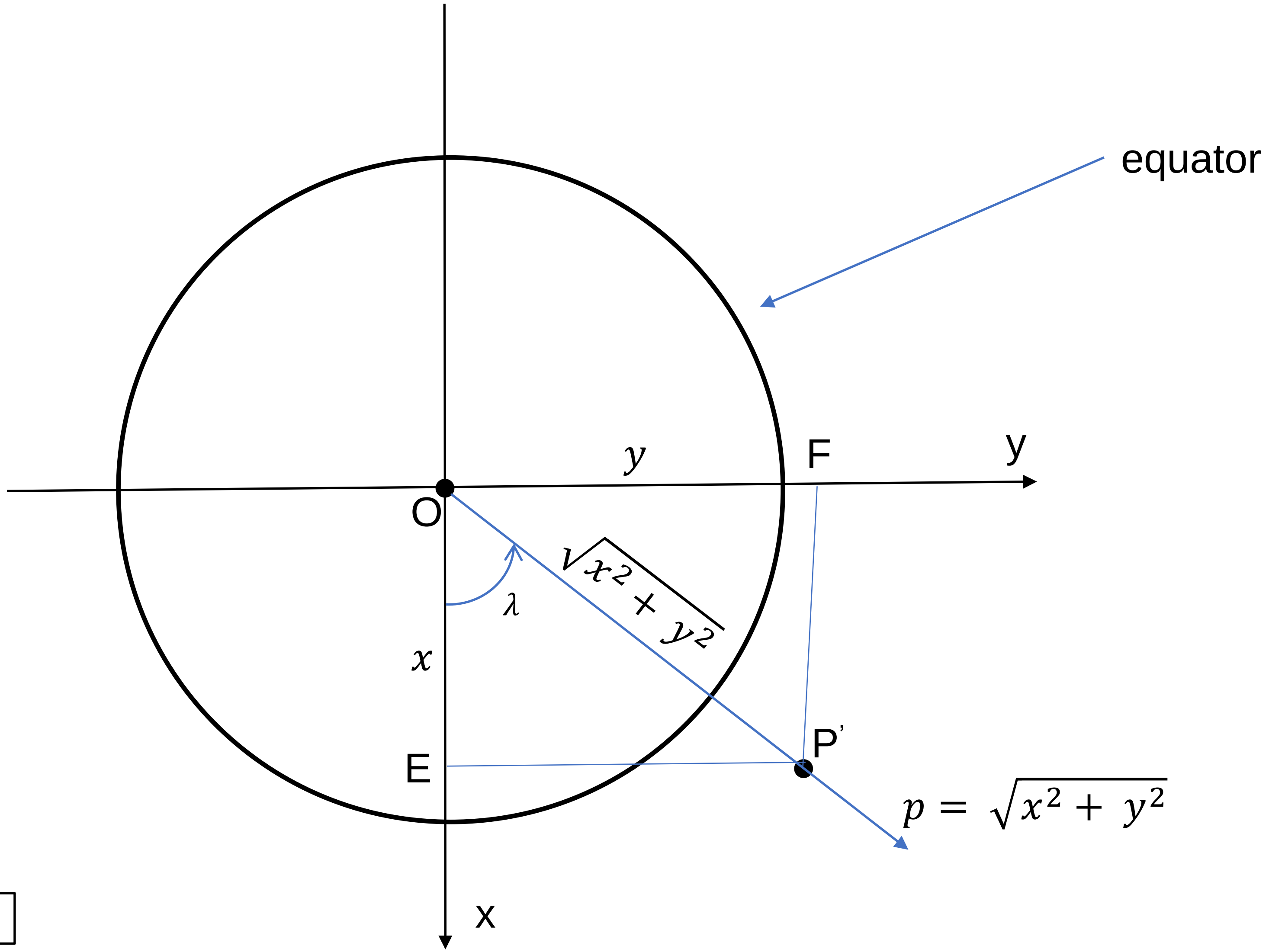
$$e \;=\; \sqrt{f\,(\,2 - f\,)}$$



a	Length of Semi-major Axis
b	Length of Semi-minor Axis
h	Geodetic Altitude
φ	Geodetic Latitude
θ	Geocentric Latitude
ψ	Reduced or Parametric Latitude

Summary

P' on equatorial plane



$$x = p\cos(\lambda)$$
$$y = p\sin(\lambda)$$

λ	**Geocentric Longitude**

Summary

φ	Geodetic Latitude
θ	Geocentric Latitude
ψ	Reduced or Parametric Latitude
λ	Geocentric Longitude

$e = $ Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$

$e_c = $ Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \frac{b}{a}$

$f = $ Elliptical Flattening Factor $= \frac{a-b}{a} = 1 - e_c$

$e = \sqrt{f\,(2 - f)}$

$\boxed{\boldsymbol{\nu} = \mathbf{N} = \textbf{Length } \overline{\mathbf{QH}} = \frac{a}{W}}$, where: $W = \sqrt{1 - e^2[\sin(\varphi)]^2}$

h $=$ Length $\overline{QP}$ = geodetic altitude

Length $\overline{OH}$ = Length $\overline{P'S}$

Length $\overline{OP'}$ = Length $\overline{HS}$

Length $\overline{DH}$ = Length $\overline{QH}$ - Length $\overline{QD}$ $= \frac{a}{W} - \frac{a}{W}(1 - e^2) = \frac{a}{W} e^2$

Length $\overline{QD}$ = Length $\overline{QH}$ − Length $\overline{DH}$ $= \frac{a}{W} - \frac{a}{W} e^2 = \frac{a}{W}(1 - e^2)$

Length $\overline{OH}$ = Length $\overline{DH}$ $\sin(\varphi) = \frac{a}{W} e^2 \sin(\varphi)$

$$\boxed{\tan(\theta) = \left(1 - e^2\right)\tan(\varphi) = \frac{b^2}{a^2}\tan(\varphi) = (1 - f)^2 \tan(\varphi)}$$

$$\boxed{\tan(\psi) = \frac{b}{a}\tan(\varphi) = \sqrt{1 - e^2}\,\tan(\varphi) = (1 - f)\tan(\varphi)}$$

Length $\overline{PH}$ = Length $\overline{QH}$ + Length $\overline{QP}$ $= \nu + h$

Length $\overline{PD}$ = Length $\overline{PQ}$ + Length $\overline{QD}$ $= h + \frac{a}{W}(1 - e^2)$

Length $\overline{DP'}$ = Length $\overline{PD}$ $cos(\varphi)$

Length $\overline{OP'}$ = Length $\overline{HS}$ = Length $\overline{PH}$ $\cos(\varphi) = (\nu + h)\cos(\varphi)$

Length $\overline{PP'}$ = Length $\overline{PS}$ − Length $\overline{P'S}$ = Length $\overline{PS}$ − Length $\overline{OH}$ = Length $\overline{PH}$ $sin(\varphi)$ − Length $\overline{DH}$ $\sin(\varphi)$

$= (\nu + h)\, sin(\varphi) -$ Length $\overline{DH}$ $\sin(\varphi) = (\nu + h)\sin(\varphi) - \frac{a}{W} e^2 \sin(\varphi) = (\nu + h)\sin(\varphi) - \nu\, e^2 \sin(\varphi) = [\,\nu\,(1 - e^2) + h\,]\sin(\varphi)$

Length $\overline{OE}$ = Length $\overline{OP'}$ $\cos(\lambda) = (\nu + h)\cos(\varphi)\cos(\lambda)$

Length $\overline{OF}$ = Length $\overline{OP'}$ $sin(\lambda) = (\nu + h)\cos(\varphi)\sin(\lambda)$

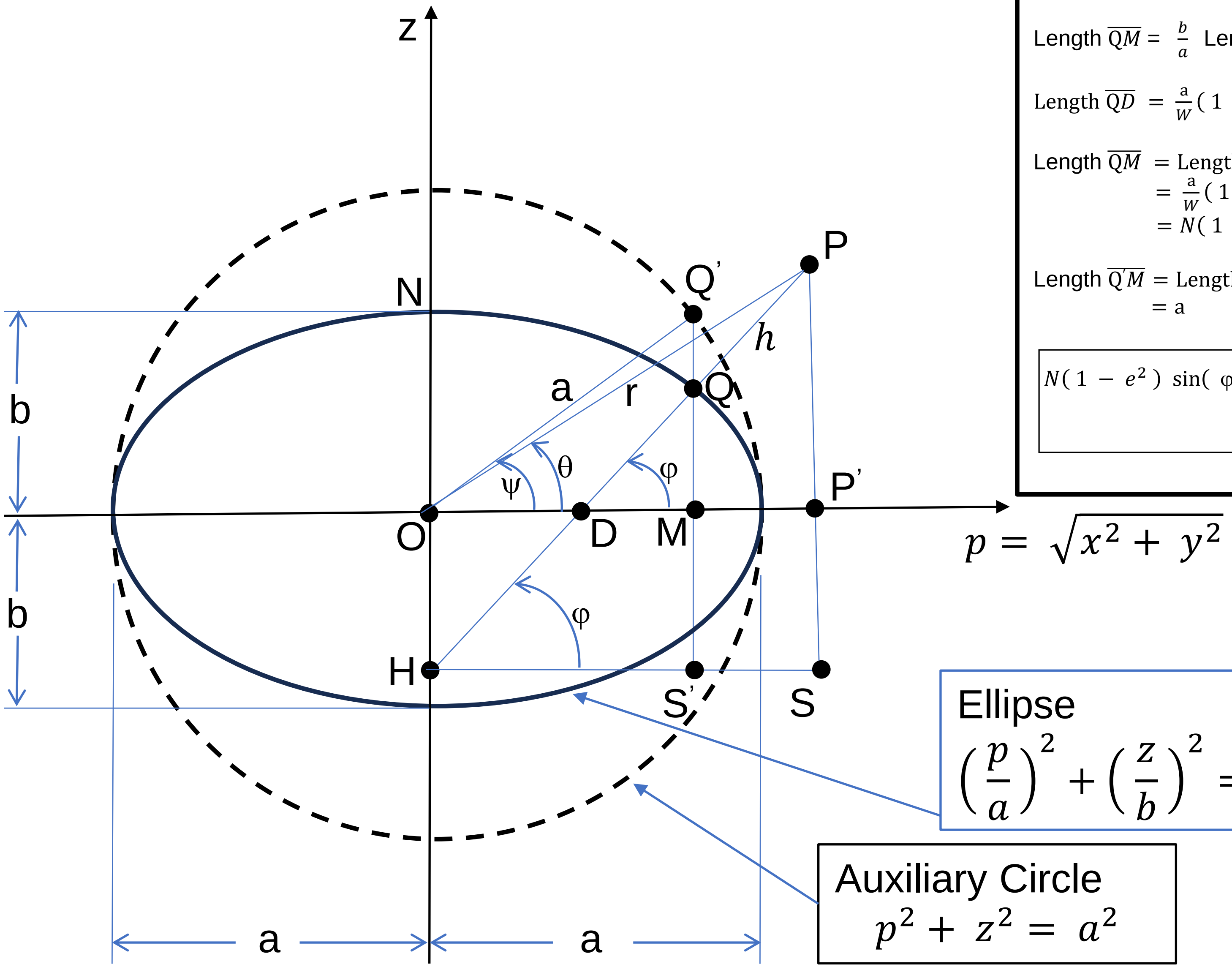
Summary

Length $\overline{HS'}$ = Length $\overline{OM}$

Length $\overline{HS'}$ = Length $\overline{QH}$ $\cos(\varphi)$
Length $\overline{OM}$ = Length $\overline{OQ'}$ $\cos(\psi)$

Length $\overline{QH}$ $= N = \frac{a}{W}$
Length $\overline{OQ'}$ $= a$

$$\boxed{N\cos(\varphi) = a\cos(\psi)}$$



Length $\overline{QM}$ = $\frac{b}{a}$ Length $\overline{Q'M}$

Length $\overline{QD}$ $= \frac{a}{W}(1 - e^2)$

Length $\overline{QM}$ = Length $\overline{QD}$ $\sin(\varphi)$
$= \frac{a}{W}(1 - e^2)\sin(\varphi)$
$= N(1 - e^2)\sin(\varphi)$

Length $\overline{Q'M}$ = Length $\overline{Q'O}$ $\sin(\psi)$
$= a \sin(\psi)$

$$\boxed{\begin{aligned} N(1 - e^2)\sin(\varphi) &= \frac{b}{a}\, a \sin(\psi) \\ &= b\sin(\psi) \end{aligned}}$$

Ellipse
$$\left(\frac{p}{a}\right)^2 + \left(\frac{z}{b}\right)^2 = 1$$

Auxiliary Circle
$$p^2 + z^2 = a^2$$

a	Length of Semi-major Axis
b	Length of Semi-minor Axis
h	Geodetic Altitude
θ	Geocentric Latitude
φ	Geodetic Latitude
ψ	Reduced or Parametric Latitude

Geodetic Coordinates { φ λ h } Converted To Cartesian Coordinates { x y z }

Given:

φ	Geodetic Latitude
λ	Geocentric Longitude
h	Geodetic Altitude

$\nu = \frac{\mathrm{a}}{W}$

$\mathrm{W} = \sqrt{1 - \mathrm{e}^2 \sin^2(\varphi)}$

$\mathrm{a} = \text{Length of Ellipse SemiMajor Axis}$

$\mathrm{b} = \text{Length of Ellipse SemiMinor Axis}$

$\mathrm{e} = \text{Ellipse Eccentricity} = \frac{\sqrt{a^2 - b^2}}{a}$

$\mathrm{e_c} = \text{Complementary Ellipse Eccentricity} = \sqrt{1 - \mathrm{e}^2} = \frac{\mathrm{b}}{\mathrm{a}}$

$\text{Length } \overline{\mathrm{P}P'} = [\, \nu \,(1 - e^2) + h \,] \sin(\varphi)$

$\text{Length } \overline{OE} = \text{Length } \overline{\mathrm{O}P'} \cos(\lambda) = (\nu + \mathrm{h}) \cos(\varphi) \cos(\lambda)$

$\text{Length } \overline{OF} = \text{Length } \overline{\mathrm{O}P'} \; sin(\lambda) = (\nu + \mathrm{h}) \cos(\varphi) \sin(\lambda)$

$$\boxed{\mathbf{x} = \textbf{Length } \overline{\boldsymbol{OE}} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \cos(\boldsymbol{\lambda})}$$

$$\boxed{\mathbf{y} = \textbf{Length } \overline{\boldsymbol{OF}} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \sin(\boldsymbol{\lambda})}$$

$$\boxed{\mathbf{z} = \textbf{Length } \overline{\mathbf{P}\boldsymbol{P'}} = \left[\, \boldsymbol{\nu} \left(\mathbf{1} - \boldsymbol{e^2} \right) + \boldsymbol{h} \,\right] \sin(\varphi)}$$

Basic Geodetic Equations

Given:	φ	Geodetic Latitude
	λ	Geocentric Longitude
	h	Geodetic Altitude
	θ	Geocentric Latitude

$\nu = \frac{a}{W}$ $\qquad$ $W = \sqrt{1 - e^2 \sin^2(\varphi)}$

a = Length of Ellipse SemiMajor Axis $\qquad$ b = Length of Ellipse SemiMinor Axis

e = Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$ $\qquad$ e_c = Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \frac{b}{a}$

Length $\overline{PP'} = [\, \nu\, (1 - e^2) + h \,]\, \sin(\varphi)$

Length $\overline{OE}$ = Length $\overline{OP'}\, \cos(\lambda) = (\nu + h)\, \cos(\varphi)\, \cos(\lambda)$

Length $\overline{OF}$ = Length $\overline{OP'}\, \sin(\lambda) = (\nu + h)\, \cos(\varphi)\, \sin(\lambda)$

$$\boxed{\mathbf{x} = \textbf{Length } \overline{\boldsymbol{OE}} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]\, \cos(\varphi)\, \cos(\boldsymbol{\lambda})}$$

$$\boxed{\mathbf{y} = \textbf{Length } \overline{\boldsymbol{OF}} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]\, \cos(\varphi)\, \sin(\boldsymbol{\lambda})}$$

$$\boxed{\mathbf{z} = \textbf{Length } \overline{\mathbf{P}\boldsymbol{P'}} = \left[\, \boldsymbol{\nu}\, \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) + \boldsymbol{h} \,\right]\, \sin(\varphi)}$$

$$\mathbf{r}\cos(\theta) = \boldsymbol{p} = \sqrt{\boldsymbol{x}^2 + \boldsymbol{y}^2} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]\, \cos(\varphi) \sqrt{\left(\cos(\boldsymbol{\lambda})\right)^2 + \left(\sin(\boldsymbol{\lambda})\right)^2} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]\, \cos(\varphi)$$

$$\mathbf{r}\sin(\theta) = \boldsymbol{z} = \textbf{Length } \overline{\mathbf{P}\boldsymbol{P'}} = \left[\, \boldsymbol{\nu}\, \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) + \boldsymbol{h} \,\right]\, \sin(\varphi)$$

$$\boxed{\mathbf{r}\cos(\theta) = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]\, \cos(\varphi)}$$

$$\boxed{\mathbf{r}\sin(\theta) = \left[\, \boldsymbol{\nu} \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) + \boldsymbol{h} \,\right]\, \sin(\varphi)}$$

Geodetic Altitude h in terms of Latitudes φ and θ (alternate form)

Given:	φ	Geodetic Latitude
	λ	Geocentric Longitude
	h	Geodetic Altitude
	θ	Geocentric Latitude

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

a = Length of Ellipse SemiMajor Axis $\quad$ b = Length of Ellipse SemiMinor Axis

e = Ellipse Eccentricity = $\frac{\sqrt{a^2 - b^2}}{a}$ $\quad$ e_c = Complementary Ellipse Eccentricity = $\sqrt{1 - e^2} = \frac{b}{a}$

$$\mathbf{r\cos(\theta) = [\ \boldsymbol{\nu} + \boldsymbol{h}\ ]\cos(\varphi) = \boldsymbol{\nu}\cos(\varphi) + \boldsymbol{h}\cos(\varphi)}$$
$$\mathbf{r\sin(\theta) = \left[\ \boldsymbol{\nu}\left(1 - \boldsymbol{e}^2\right) + \boldsymbol{h}\ \right]\sin(\varphi) = \boldsymbol{\nu}\left(1 - \boldsymbol{e}^2\right)\sin(\varphi) + \boldsymbol{h}\sin(\varphi)}$$

$$h\cos(\varphi) = r\cos(\theta) - \nu\cos(\varphi) \qquad h\cos^2(\varphi) = r\cos(\theta)\cos(\varphi) - \nu\cos^2(\varphi) \qquad h\cos^2(\varphi) = r\cos(\theta)\cos(\varphi) - \nu\left(1 - sin^2(\varphi)\right)$$
$$h\sin(\varphi) = r\sin(\theta) - \nu(1 - e^2)\sin(\varphi) \qquad h\sin^2(\varphi) = r\sin(\theta)\sin(\varphi) - \nu(1 - e^2)\sin^2(\varphi) \qquad h\sin^2(\varphi) = r\sin(\theta)\sin(\varphi) - \nu(1 - e^2)\sin^2(\varphi)$$

$$h\,[\cos^2(\varphi) + \sin^2(\varphi)\,] = r\ \{\ [\cos(\theta)\ \cos(\varphi)] + [\sin(\theta)\ \sin(\varphi)]\ \} - \nu\ \{\ 1 - \sin^2(\varphi) + (1 - e^2)\sin^2(\varphi)\ \}$$
$$h = r\ \{\ [\cos(\theta)\ \cos(\varphi)] + [\sin(\theta)\ \sin(\varphi)]\ \} - \nu\ \{\ 1 - e^2\sin^2(\varphi)\ \}$$
$$h = r\ \cos(\ \theta - \varphi\ ) - \nu\ \{\ 1 - e^2\sin^2(\varphi)\ \} = r\ co(\ \theta - \varphi\ ) - \nu\ \frac{a^2}{\nu^2} = r\ \cos(\ \theta - \varphi\ ) - \frac{a^2}{\nu}$$

Thus:

$$\boxed{\boldsymbol{h} = \mathbf{r}\ \mathbf{\cos}(\ \theta - \varphi\ ) - \frac{\boldsymbol{a}^2}{\boldsymbol{\nu}}}$$

Geodetic Latitude φ in terms of Geocentric Latitude θ (alternate form)

Given: φ Geodetic Latitude
λ Geocentric Longitude
h Geodetic Altitude
θ Geocentric Latitude

$\nu = \frac{a}{W}$ $\qquad$ $W = \sqrt{1 - e^2 \sin^2(\chi)}$

$a =$ Length of Ellipse SemiMajor Axis $\qquad$ $b =$ Length of Ellipse SemiMinor Axis

$e =$ Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$ $\qquad$ $e_c =$ Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \frac{b}{a}$

$$\mathrm{r}\cos(\theta) = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) = \boldsymbol{\nu} \cos(\varphi) + \boldsymbol{h}\cos(\varphi)$$
$$\mathrm{r}\sin(\theta) = \left[\, \boldsymbol{\nu}\left(1 - \boldsymbol{e}^2\right) + \boldsymbol{h} \,\right] \sin(\varphi) = \boldsymbol{\nu}\left(1 - \boldsymbol{e}^2\right)\sin(\varphi) + \boldsymbol{h}\sin(\varphi)$$

$$\mathrm{r}\cos(\theta)\sin(\varphi) = \nu \cos(\varphi)\sin(\varphi) + h\cos(\varphi)\sin(\varphi)$$
$$\mathrm{r}\sin(\theta)\cos(\varphi) = \nu(1 - e^2)\sin(\varphi)\cos(\varphi) + h\sin(\varphi)\cos(\varphi)$$

Eliminate the geodetic altitude h:

$$\mathrm{r}\,[\cos(\theta)\sin(\varphi) - \sin(\theta)\cos(\varphi)] = \nu\,[\,1 - (1 - e^2)\,]\cos(\varphi)\sin(\varphi) = \nu\, e^2\cos(\varphi)\sin(\varphi)$$

$$\mathrm{r}\sin(\theta - \varphi) = \nu\, e^2\cos(\varphi)\sin(\varphi)$$

$$\boxed{\sin(\varphi - \theta) = \frac{\boldsymbol{\nu}\, \boldsymbol{e}^2 \cos(\varphi)\sin(\varphi)}{\boldsymbol{r}}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Geocentric Longitude λ

Given:

x	Rectangular coordinate x
y	Rectangular coordinate y
z	Rectangular coordinate z

$\nu = \frac{\mathrm{a}}{W}$

$\mathrm{W} = \sqrt{1 - \mathrm{e}^2 \sin^2(\varphi)}$

a = Length of Ellipse SemiMajor Axis

b = Length of Ellipse SemiMinor Axis

e = Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$

$\mathrm{e_c}$ = Complementary Ellipse Eccentricity $= \sqrt{1 - \mathrm{e}^2} = \frac{\mathrm{b}}{\mathrm{a}}$

$$\left.\begin{aligned} \boldsymbol{x} &= [\,\boldsymbol{\nu} + \boldsymbol{h}\,] \cos(\varphi) \cos(\boldsymbol{\lambda}) \\ \mathbf{y} &= [\,\boldsymbol{\nu} + \boldsymbol{h}\,] \cos(\varphi) \sin(\boldsymbol{\lambda}) \end{aligned}\right\} \rightarrow \left\{\begin{aligned} \frac{\boldsymbol{x}}{\cos(\boldsymbol{\lambda})} &= [\,\boldsymbol{\nu} + \boldsymbol{h}\,] \cos(\varphi) \\ \frac{\boldsymbol{y}}{\sin(\boldsymbol{\lambda})} &= [\,\boldsymbol{\nu} + \boldsymbol{h}\,] \cos(\varphi) \end{aligned}\right\} \rightarrow \boxed{\frac{\boldsymbol{x}}{\cos(\boldsymbol{\lambda})} = \frac{\boldsymbol{y}}{\sin(\boldsymbol{\lambda})}}$$

$$\left.\begin{aligned} \boldsymbol{x} &= \boldsymbol{p} \cos(\boldsymbol{\lambda}) \\ \mathbf{y} &= \boldsymbol{p} \sin(\boldsymbol{\lambda}) \end{aligned}\right\} \rightarrow \boxed{\frac{\boldsymbol{x}}{\cos(\boldsymbol{\lambda})} = \frac{\boldsymbol{y}}{\sin(\boldsymbol{\lambda})} = \boldsymbol{p}} \rightarrow \frac{\boldsymbol{y}}{\boldsymbol{x}} = \frac{\sin(\boldsymbol{\lambda})}{\cos(\boldsymbol{\lambda})} = \tan(\boldsymbol{\lambda})$$

Thus the Geocentric Longitude is determined as: $\boxed{\boldsymbol{\lambda} = \mathbf{atan2}(\boldsymbol{y},\ \boldsymbol{x})}$

NOTE(s):

- Sometimes the arctangent function is written using the half angle formula as:

  $$\frac{y}{x} = \frac{\sin(\lambda)}{\cos(\lambda)} = \tan(\lambda) \quad \text{so} \quad \tan\left(\frac{1}{2}\lambda\right) = \frac{\sin(\lambda)}{\cos(\lambda) + 1} = \frac{\sin(\lambda)}{\cos(\lambda) + \sqrt{cos^2(\lambda) + sin^2(\lambda)}} = \frac{y}{x + \sqrt{x^2 + y^2}} \quad \text{thus} \quad \boxed{\lambda = 2\ atan\left(\frac{y}{x + \sqrt{x^2 + y^2}}\right)}$$

  This allows obtaining a four quadrant value for the angle λ.
- Using sufficiently recent math libraries will allow obtaining the four quadrant value for the angle λ via $\lambda = atan2(\,y,\ x\,)$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Geodetic Altitude h   Given Geodetic Latitude φ

Given:

φ Geodetic Latitude

$\nu = \frac{\mathrm{a}}{W}$

$\mathrm{W} = \sqrt{1 - \mathrm{e}^2 \sin^2(\varphi)}$

$\mathrm{a} =$ Length of Ellipse SemiMajor Axis

$\mathrm{b} =$ Length of Ellipse SemiMinor Axis

$\mathrm{e} =$ Ellipse Eccentricity $= \frac{\sqrt{\mathrm{a}^2 - \mathrm{b}^2}}{\mathrm{a}}$

$\mathrm{e_c} =$ Complementary Ellipse Eccentricity $= \sqrt{1 - \mathrm{e}^2} = \frac{\mathrm{b}}{\mathrm{a}}$

$$\boldsymbol{x} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \cos(\boldsymbol{\lambda}) \rightarrow \boldsymbol{p} = \frac{\boldsymbol{x}}{\cos(\boldsymbol{\lambda})} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \rightarrow \frac{\boldsymbol{p}}{\cos(\varphi)} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \rightarrow \boldsymbol{h} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \boldsymbol{\nu} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \frac{\mathbf{a}}{\sqrt{1 - e^2 \, sin^2(\varphi)}}$$

Thus the Geodetic Altitude can be detemined as:

$$\boxed{\boldsymbol{h} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \frac{\mathbf{a}}{\sqrt{1 - e^2 \sin^2(\varphi)}}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Geodetic Equations Given Cartesian Coordinates { x y z }

Given:

x	Rectangular coordinate x
y	Rectangular coordinate y
z	Rectangular coordinate z

$\nu = \frac{a}{W}$

$W = \sqrt{1 - e^2 \sin^2(\varphi)}$

$a = $ Length of Ellipse SemiMajor Axis

$b = $ Length of Ellipse SemiMinor Axis

$e = $ Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$

$e_c = $ Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \frac{b}{a}$

The Geodetic Altitude is: $\boxed{\boldsymbol{h} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \frac{\mathbf{a}}{\sqrt{1 - e^2 \sin^2(\varphi)}} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \frac{\mathbf{a}}{W} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \nu}$

The Rectangular Z Coordinate is: $\boxed{|\mathbf{z}| = \textbf{Length } \overline{\mathbf{P}P'} = [\, \boldsymbol{\nu}\,(1 - \boldsymbol{e}^2) + \boldsymbol{h}\,] \sin(\varphi)}$

$$\rightarrow |\mathbf{z}| = \left[\boldsymbol{\nu}\,(1 - \boldsymbol{e}^2) + \frac{\boldsymbol{p}}{\cos(\varphi)} - \nu \right] \sin(\varphi)$$

$$|\mathbf{z}| = \left[\boldsymbol{\nu}\,(1 - \boldsymbol{e}^2) + \frac{\boldsymbol{p}}{\cos(\varphi)} - \nu \right] \sin(\varphi) = \left[-\boldsymbol{\nu}\,\boldsymbol{e}^2 + \frac{\boldsymbol{p}}{\cos(\varphi)} \right] \sin(\varphi) = \frac{\boldsymbol{p}\,\sin(\varphi)}{\cos(\varphi)} - \boldsymbol{\nu}\,\boldsymbol{e}^2 \sin(\varphi) = \boldsymbol{p}\;\boldsymbol{tan}(\varphi) - \boldsymbol{\nu}\,\boldsymbol{e}^2 \sin(\varphi)$$

$$\boxed{\boldsymbol{p}\;\boldsymbol{tan}(\varphi) - \boldsymbol{\nu}\,\boldsymbol{e}^2 \sin(\varphi) - |\mathbf{z}| = \mathbf{0}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Geodetic Equations Given Cartesian Coordinates { x y z }

Given:

x	Rectangular coordinate x
y	Rectangular coordinate y
z	Rectangular coordinate z

$\nu = \frac{a}{W}$

$W = \sqrt{1 - e^2 \sin^2(\varphi)}$

$a =$ Length of Ellipse SemiMajor Axis

$b =$ Length of Ellipse SemiMinor Axis

$e =$ Ellipse Eccentricity $= \frac{\sqrt{a^2 - b^2}}{a}$

$e_c =$ Complementary Ellipse Eccentricity $= \sqrt{1 - e^2} = \frac{b}{a}$

$\boldsymbol{p} = \sqrt{x^2 + y^2}$

Non-Linear Transcendental Equations For Geodetics { φ λ h } Given Cartesian Rectangular Coordinates { x y z }

$$(1) \quad \boxed{\boldsymbol{\lambda} = \mathrm{atan2}(\boldsymbol{y}, \boldsymbol{x})}$$

$$(2) \quad \boxed{\boldsymbol{p} \; \boldsymbol{tan}(\varphi) - \boldsymbol{\nu} \, \boldsymbol{e}^2 \sin(\varphi) - |\, z \,| = 0}$$

$$(3) \quad \boxed{\boldsymbol{h} = \frac{\boldsymbol{p}}{\cos(\varphi)} - \frac{a}{\sqrt{1 - \boldsymbol{e}^2 \, \boldsymbol{sin}^2(\varphi)}}}$$

Strategy:

1. Use Equation (1) for Geocentric Longitude $\boldsymbol{\lambda}$ Given $\boldsymbol{x}$, $\mathbf{y}$ **(Easy)**
2. Solve Equation (2) for Geodetic Latitude φ Given $\boldsymbol{x}$, $\mathbf{y}$, $\mathbf{z}$ **(Very Hard)**
3. Use Equation (3) for Geodetic Altitude $\boldsymbol{h}$ Given φ **(Easy)**

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Relationships Between Geodetic Latitude φ and Reduced Latitude ψ

$$\nu = \frac{a}{W}$$
$$W = \sqrt{1 - e^2 \sin^2(\varphi)}$$
$$a = \text{Length of Ellipse SemiMajor Axis}$$
$$b = \text{Length of Ellipse SemiMinor Axis}$$
$$e = \text{Ellipse Eccentricity} = \frac{\sqrt{a^2 - b^2}}{a}$$
$$1 - e^2 = 1 - \frac{a^2 - b^2}{a^2} = \frac{a^2 - (a^2 - b^2)}{a^2} = \frac{b^2}{a^2}$$
$$e_c = \text{Complementary Ellipse Eccentricity} = \sqrt{1 - e^2} = \frac{b}{a}$$

1. $\tan(\psi) = \frac{b}{a}\tan(\varphi) = \sqrt{1 - e^2}\,\tan(\varphi) = e_c\,\tan(\varphi) = (1 - f)\,\tan(\varphi)$
2. $\nu\,\cos(\varphi) = a\,\cos(\psi)$
3. $\nu\,(1 - e^2)\,\sin(\varphi) = b\,\sin(\psi)$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Transform Geodetic Latitude φ Equation

Starting with: $\boxed{\mathbf{p}\ \tan(\varphi) - \nu\, e^2 \sin(\varphi) - |\,\mathbf{z}\,| = 0}$

Let T = tan($\boldsymbol{\psi}$) where $\boldsymbol{\psi}$ is the reduced latitude. $\tan(\boldsymbol{\psi}) = \frac{\mathbf{b}}{\mathbf{a}} \tan(\varphi)$ or $\tan(\varphi) = \frac{\mathbf{a}}{\mathbf{b}} \tan(\boldsymbol{\psi}) = \frac{1}{\frac{\mathbf{b}}{\mathbf{a}}} \tan(\boldsymbol{\psi}) = \frac{1}{e_c} \tan(\boldsymbol{\psi})$

$$0 = \mathbf{p}\ \tan(\varphi) - \nu\, e^2 \sin(\varphi) - |\,\mathbf{z}\,| = \mathbf{p}\ \left[\frac{\mathbf{a}}{\mathbf{b}} \tan(\boldsymbol{\psi})\right] - \nu\, e^2 \sin(\varphi) - |\,\mathbf{z}\,| = \mathbf{p}\ \left[\frac{\mathbf{a}}{\mathbf{b}}\ \mathbf{T}\right] - \nu\, e^2 \sin(\varphi) - |\,\mathbf{z}\,|$$

Then: $0 = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{b}}{\mathbf{a}}\, \nu\, e^2 \sin(\varphi) - \frac{\mathbf{b}}{\mathbf{a}} |\,\mathbf{z}\,|$

$$\sin(\varphi) = \frac{\mathrm{b}}{\nu\,(1 - \mathrm{e}^2)} \sin(\boldsymbol{\psi})$$

$$0 = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{b}}{\mathbf{a}}\, \nu\, e^2 \sin(\varphi) - \frac{\mathbf{b}}{\mathbf{a}} |\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{b}}{\mathbf{a}}\, \nu\, e^2\, \frac{\mathrm{b}}{\nu\,(1 - \mathrm{e}^2)} \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}} |\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{b}^2}{\mathbf{a}}\, \frac{e^2}{(1 - \mathrm{e}^2)} \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}} |\,\mathbf{z}\,|$$

So: $$\boxed{0 = \mathbf{p}\ T - \frac{\mathbf{b}^2}{\mathbf{a}}\, \frac{e^2}{(1 - e^2)}\ \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}} |\,\mathbf{z}\,|}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }

Transform Geodetic Latitude φ Equation

$$\mathbf{0} = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{b^2}}{\mathbf{a}}\ \frac{a^2 - b^2}{b^2}\ \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}\,|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \frac{a^2 - b^2}{a}\ \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}\,|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \frac{a^2 - b^2}{a^2}\ \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}\,|\,\mathbf{z}\,|$$

Now: $e\ =\ \text{Ellipse Eccentricity} = \frac{\sqrt{a^2 - b^2}}{a}$

So: $\mathbf{0} = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \sin(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \sin(\boldsymbol{\psi})\ \frac{\cos(\boldsymbol{\psi})}{\cos(\boldsymbol{\psi})} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \frac{\sin(\boldsymbol{\psi})}{\cos(\boldsymbol{\psi})}\cos(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \tan(\boldsymbol{\psi})\ \cos(\boldsymbol{\psi}) - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,|$

$$= \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \tan(\boldsymbol{\psi})\ \frac{1}{\frac{1}{\cos(\boldsymbol{\psi})}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \tan(\boldsymbol{\psi})\ \frac{1}{\frac{1}{\sqrt{\cos^2(\boldsymbol{\psi})}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \tan(\boldsymbol{\psi})\ \frac{1}{\sqrt{\frac{\cos^2(\boldsymbol{\psi}) + sin^2(\boldsymbol{\psi})}{\cos^2(\boldsymbol{\psi})}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,|$$

$$= \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}\ \frac{1}{\sqrt{\frac{\cos^2(\boldsymbol{\psi}) + sin^2(\boldsymbol{\psi})}{\cos^2(\boldsymbol{\psi})}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}\ \frac{1}{\sqrt{1 + \frac{sin^2(\boldsymbol{\psi})}{\cos^2(\boldsymbol{\psi})}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}\ \frac{1}{\sqrt{1 + tan^2(\boldsymbol{\psi})}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,|$$

$$= \mathbf{p}\ \mathbf{T} - \mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}\ \frac{1}{\sqrt{1 + \mathbf{T^2}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}}{\sqrt{1 + \mathbf{T^2}}} - \frac{\mathbf{b}}{\mathbf{a}}|\,\mathbf{z}\,| = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}}{\sqrt{1 + \mathbf{T^2}}} - \sqrt{1 - e^2}\ |\,\mathbf{z}\,|$$

So: $\boxed{0 = \mathbf{p}\ \mathbf{T} - \frac{\mathbf{a}\ \boldsymbol{e^2}\ \mathrm{T}}{\sqrt{1 + \mathbf{T^2}}} - \sqrt{1 - e^2}\ |\,\mathbf{z}\,|}$ Then: $0 = \left(\frac{\mathbf{p}}{a}\right)\ \mathbf{T} - \frac{\boldsymbol{e^2}\ \mathrm{T}}{\sqrt{1 + \mathbf{T^2}}} - \left(\frac{\sqrt{1 - e^2}}{a}\right)|\,\mathbf{z}\,|$

$\mathrm{T} = \tan(\,\boldsymbol{\psi}\,)$

Let: $\mathrm{P} = \frac{\mathbf{p}}{a}$ $\mathrm{E} = \boldsymbol{e^2}$ $\mathrm{Z} = \frac{\sqrt{1 - e^2}}{a}\ |\ z\ |$ So: : $\boxed{\mathbf{P}\,\mathbf{T} - \frac{\mathbf{E}\ \mathbf{T}}{\sqrt{\mathbf{1} + \mathbf{T^2}}} - \mathbf{Z}\ =\ \mathbf{0}}$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Latitude φ Equation

Equation To Be Solved: $\boxed{0 = P\ \mathbf{T} - \frac{\mathbf{E}\ \mathrm{T}}{\sqrt{1 + \mathbf{T}^2}} - Z}$ With: ψ = Reduced Latitude $\mathrm{T} = \tan(\psi) = e_c \tan(\varphi)$ $P = \frac{\mathbf{p}}{a}$ $E = \boldsymbol{e^2}$ $Z = \frac{\sqrt{1 - e^2}}{a} \mid z \mid$

Approaches:

1. Compute Direct Solution Of Equation
   a. Use of Ferrari's Method
      Transform the equation to be solved into a fourth order nonlinear algebraic equation.
      Use Ferrari's Method to compute a direct solution of the fourth order nonlinear algebraic equation.
2. Iterative Numerical Approximation To Solution Of Equation
   a. Newton's Second Order Method
      Compute iterative approximations for the solution of the equation using the second order Newton's Method
   b. Halley's Third Order Method
      Compute iterative approximations for the solution of the equation using the third order Halley's Method

NOTE(s):

1. Normally direct methods are preferred over iterative methods.
2. The Ferrari direct method is used to solve fourth order (quartic) non-linear algebraic equations.
   Ferrari's method is very complicated.
   It involves several transcendental function calls (sqrt, cube root, etc.)
3. The iterative Newton and Halley methods can be faster than the Ferrari direct method if sufficient accuracy can be obtained using only a very few (ie one) iteration(s).

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Latitude φ Equation

Equation To Be Solved: $\boxed{0 = \text{P}\ \mathbf{T} - \dfrac{\text{E T}}{\sqrt{1+\mathbf{T}^2}} - \text{Z}}$ With: ψ = Reduced Latitude $\quad \text{T} = \tan(\psi) = e_c \tan(\varphi) \quad \text{P} = \frac{\text{p}}{\text{a}} \quad \text{E} = e^2 \quad \text{Z} = \frac{\sqrt{1-\text{e}^2}}{\text{a}} \mid z \mid$

Approach: 2b) Halley's Third Order Method
Compute iterative approximations for the solution of the equation using the third order Halley's Method

Define: $\boxed{\text{g}(\ \mathbf{T}\ ) = \text{P}\ \mathbf{T} - \dfrac{\text{E T}}{\sqrt{1+\mathbf{T}^2}} - \text{Z}}$ Then the equation to be solved is: $\text{g}(\ \mathbf{T}\ ) = 0$

$$\text{g}'(\ \mathbf{T}\ ) = \frac{d}{dT}\text{g}(\ \mathbf{T}\ ) = \frac{d}{dT}\left[\text{P}\ \mathbf{T} - \frac{\text{E T}}{\sqrt{1+\mathbf{T}^2}} - \text{Z}\right] = \text{P} - \frac{1}{(1+\mathbf{T}^2)}\left\{E\sqrt{1+\mathbf{T}^2} - (ET)\left(\frac{1}{2}\right)\left(\frac{2T}{\sqrt{1+\mathbf{T}^2}}\right)\right\} = \text{P} - \frac{1}{(1+\mathbf{T}^2)}\left\{\frac{\text{E}(1+\mathbf{T}^2) - \text{E}\,\mathbf{T}^2}{\sqrt{1+\mathbf{T}^2}}\right\}$$

$$= \text{P} - \frac{1}{(1+\mathbf{T}^2)}\left\{\frac{\text{E} + \text{E}\,\mathbf{T}^2 - \text{E}\,\mathbf{T}^2}{\sqrt{1+\mathbf{T}^2}}\right\} = \text{P} - \frac{1}{(1+\mathbf{T}^2)}\left\{\frac{\text{E}}{\sqrt{1+\mathbf{T}^2}}\right\} = \text{P} - \frac{\text{E}}{(1+\mathbf{T}^2)^{\frac{3}{2}}}$$

$$\boxed{\text{g}'(\ \mathbf{T}\ ) = \text{P} - \frac{\text{E}}{(1+\mathbf{T}^2)^{\frac{3}{2}}}}$$

$$\text{g}''(\ \mathbf{T}\ ) = \frac{d}{dT}\text{g}'(\ \mathbf{T}\ ) = \frac{d}{dT}\left[\text{P} - \frac{\text{E}}{(1+\mathbf{T}^2)^{\frac{3}{2}}}\right] = -\text{E}\left\{-\frac{3}{2}\ \frac{2\,\text{T}}{(1+\mathbf{T}^2)^{\frac{5}{2}}}\right\} = \frac{3\,\text{E T}}{(1+\mathbf{T}^2)^{\frac{5}{2}}}$$

$$\boxed{\text{g}''(\ \mathbf{T}\ ) = \frac{3\,\text{E T}}{(1+\mathbf{T}^2)^{\frac{5}{2}}}}$$

NOTE: Fortunately g''(T) is evaluated with only a small increase in the computation time.

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Latitude φ Equation

Equation To Be Solved: $\boxed{0 = \mathrm{P}\ \mathbf{T} - \frac{\mathbf{E}\ \mathrm{T}}{\sqrt{1+\mathbf{T}^2}} - \mathrm{Z}}$ With: ψ = Reduced Latitude $\mathrm{T} = \tan(\psi) = e_c \tan(\varphi)$ $\mathrm{P} = \frac{\mathbf{p}}{\mathrm{a}}$ $\mathrm{E} = \boldsymbol{e}^{\mathbf{2}}$ $\mathrm{Z} = \frac{\sqrt{1-e^2}}{\mathrm{a}}\ |\ \mathrm{z}\ |$

$\boxed{\mathrm{g}(\mathbf{T}) = \mathrm{P}\ \mathbf{T} - \frac{\mathbf{E}\ \mathrm{T}}{\sqrt{1+\mathbf{T}^2}} - \mathrm{Z}}$ Then the equation to be solved is: $\mathrm{g}(\mathbf{T}) = 0$

Halley's Third Order Method

$$\mathbf{T}_{n+1} = \mathbf{T}_n - \frac{\mathrm{g}(\mathbf{T}_n)}{\mathrm{g}'(\mathbf{T}_n) - \left[\frac{\mathrm{g}''(\mathbf{T}_n)\ \mathrm{g}(\mathbf{T}_n)}{2\ \mathrm{g}'(\mathbf{T}_n)}\right]} = \mathbf{T}_n - \frac{2\ \mathrm{g}(\mathbf{T}_n)\ \mathrm{g}'(\mathbf{T}_n)}{2\left(\mathrm{g}'(\mathbf{T}_n)\right)^2 - \mathrm{g}''(\mathbf{T}_n)\ \mathrm{g}(\mathbf{T}_n)}$$

$$\boxed{\mathbf{T}_{n+1} = \mathbf{T}_n - \frac{2\ \mathrm{g}(\mathbf{T}_n)\ \mathrm{g}'(\mathbf{T}_n)}{2\left(\mathrm{g}'(\mathbf{T}_n)\right)^2 - \mathrm{g}''(\mathbf{T}_n)\ \mathrm{g}(\mathbf{T}_n)}}$$

$$\mathbf{T}_{n+1} = \mathbf{T}_n - \frac{2\ \mathrm{g}(\mathbf{T}_n)\ \mathrm{g}'(\mathbf{T}_n)}{2\left(\mathrm{g}'(\mathbf{T}_n)\right)^2 - \mathrm{g}''(\mathbf{T}_n)\ \mathrm{g}(\mathbf{T}_n)} = \mathbf{T}_n - \frac{2\left[\mathrm{P}\ \mathbf{T}_n - \frac{\mathbf{E}\ \mathbf{T}_n}{\sqrt{1+(\mathbf{T}_n)^2}} - \mathrm{Z}\right]\left[\mathrm{P} - \frac{\mathrm{E}}{(1+(\mathbf{T}_n)^2)^{\frac{3}{2}}}\right]}{2\left(\left[\mathrm{P} - \frac{\mathrm{E}}{(1+(\mathbf{T}_n)^2)^{\frac{3}{2}}}\right]\right)^2 - \left[\frac{3\ \mathrm{E}\ \mathrm{T}}{(1+(\mathbf{T}_n)^2)^{\frac{5}{2}}}\right]\left[\mathrm{P}\ \mathbf{T}_n - \frac{\mathbf{E}\ \mathbf{T}_n}{\sqrt{1+(\mathbf{T}_n)^2}} - \mathrm{Z}\right]}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Equations for p and z in terms of Geodetic Latitudes φ and Reduced Latitude ψ

$$\nu = \frac{a}{W}$$
$$W = \sqrt{1 - e^2 \sin^2(\varphi)}$$

$$\nu \cos(\varphi) = a \cos(\boldsymbol{\psi})$$
$$\nu \,(1 - e^2)\, \sin(\varphi) = b \sin(\boldsymbol{\psi})$$

$$\boldsymbol{x} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \cos(\boldsymbol{\lambda})$$
$$\boldsymbol{y} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \sin(\boldsymbol{\lambda})$$
$$\mathbf{z} = \left[\, \boldsymbol{\nu} \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) + \boldsymbol{h} \,\right] \sin(\varphi)$$

$$\boldsymbol{p}^2 = \boldsymbol{x}^2 + \boldsymbol{y}^2 = \{ \, [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \cos(\boldsymbol{\lambda}) \, \}^2 + \{ \, [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \cos(\varphi) \sin(\boldsymbol{\lambda}) \, \}^2$$
$$= [\, \boldsymbol{\nu} + \boldsymbol{h} \,]^2 \, \boldsymbol{cos}^2(\varphi) \left\{ \boldsymbol{cos}^2(\lambda) + \boldsymbol{sin}^2(\lambda) \right\} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,]^2 \, \boldsymbol{cos}^2(\varphi)$$
$$\boldsymbol{p} = [\, \boldsymbol{\nu} + \boldsymbol{h} \,] \boldsymbol{cos}(\varphi) = \boldsymbol{\nu} \, \boldsymbol{cos}(\varphi) + \boldsymbol{h} \, \boldsymbol{cos}(\varphi) = a \cos(\boldsymbol{\psi}) + \boldsymbol{h} \, \boldsymbol{cos}(\varphi)$$

$$\boxed{\boldsymbol{p} = \boldsymbol{a} \, \boldsymbol{cos}(\boldsymbol{\psi}) + \boldsymbol{h} \, \boldsymbol{cos}(\varphi)}$$

$$\mathbf{z} = \left[\, \boldsymbol{\nu} \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) + \boldsymbol{h} \,\right] \sin(\varphi) = \boldsymbol{\nu} \left(\mathbf{1} - \boldsymbol{e}^{\mathbf{2}} \right) \sin(\varphi) + \boldsymbol{h} \sin(\varphi) = \boldsymbol{b} \sin(\boldsymbol{\psi}) + \boldsymbol{h} \sin(\varphi)$$

$$\boxed{\boldsymbol{z} = \mathbf{b} \, \mathbf{sin}(\boldsymbol{\psi}) + \boldsymbol{h} \, \mathbf{sin}(\varphi)}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }

Halley's Third Order Method : Starting Value $\mathbf{T_0}$

$$\boxed{p \;= a\;\cos(\psi) + h\;\cos(\varphi)}$$

$$\boxed{|\;z\;| = \mathrm{b}\;\sin(\psi) + h\;\sin(\varphi)}$$

Obtain a starting value $\mathbf{T}_0$ for the $\mathbf{T}_{n+1}$ iteration, **assume $h = 0$ (on reference ellipsoid)**:

$$\mathrm{p} \;= \mathrm{a}\;\cos(\psi) + \mathrm{h}\;\cos(\varphi) \;= \mathrm{a}\;\cos(\psi)$$
$$|\;z\;| = \mathrm{b}\;\sin(\psi) + \mathrm{h}\;\sin(\varphi) \;= \mathrm{b}\;\sin(\psi)$$

$$\left.\begin{aligned}\sin(\psi) &= \frac{|\;z\;|}{\mathrm{b}}\\ \cos(\psi) &= \frac{\mathrm{p}}{\mathrm{a}}\end{aligned}\right\} \Rightarrow \tan(\psi) = \frac{\frac{z}{\mathrm{b}}}{\frac{\mathrm{p}}{\mathrm{a}}} = \frac{\mathrm{a}\;|\;z\;|}{\mathrm{b}\;p} \Rightarrow \;\psi \;= \mathrm{atan}\left(\frac{\mathrm{a}\;|\;z\;|}{\mathrm{b}\;p}\right)$$

So: $\psi_0 \;= \mathrm{atan}\left(\frac{\mathrm{a}\;|\;z\;|}{\mathrm{b}\;p}\right)$ and $\mathrm{T}_0 \;= \tan(\;\psi_0\;) = \tan\left(\mathrm{atan}\left(\frac{\mathrm{a}\;|\;z\;|}{\mathrm{b}\;p}\right)\right) = \frac{\mathrm{a}\;|\;z\;|}{\mathrm{b}\;p} = \frac{|\;z\;|}{p}\;\frac{\mathrm{a}}{\mathrm{b}} = \frac{|\;z\;|}{p}\;\frac{1}{\frac{\mathrm{b}}{\mathrm{a}}} = \frac{|\;z\;|}{p}\;\frac{1}{e_c} = \frac{|\;z\;|}{e_c\;p}$

$$\boxed{\mathrm{T}_0 \;= \frac{|\;z\;|}{e_c\;p}}$$

is a starting value for the Halley's Third Order Iterative Method **Assuming Zero Geodetic Altitude (h = 0)**

NOTE(s):

- In practical cases such as from the deepest ocean floor to the location of infinitely high elevations, the application of both Halley's and Newton's methods work well when the zero height solution is chosen as the starter. See Reference [1].

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Latitude φ Equation

Equation To Be Solved: $\boxed{0 = \mathrm{P}\ \mathbf{T} - \frac{\mathbf{E}\ \mathrm{T}}{\sqrt{1+\mathbf{T}^2}} - \mathrm{Z}}$ With: ψ = Reduced Latitude $\mathrm{T} = \tan(\psi) = \mathrm{e_c}\ \tan(\varphi)$ $\mathrm{P} = \frac{\mathbf{p}}{\mathrm{a}}$ $\mathrm{E} = \boldsymbol{e^2}$ $\mathrm{Z} = \frac{\sqrt{1-\mathrm{e}^2}}{\mathrm{a}}\ |\ \mathrm{z}\ | = \frac{e_c}{\mathrm{a}}\ |\ \mathrm{z}\ |$

Halley's Third Order Method

Let T_{n+1} be a sufficiently accurate solution generated by Halley's Third Order Iteration Method (typically only one Halley's iteration is necessary)

Solution For Geodetic Coordinates:

Geodetic Latitude: φ

$$\mathrm{T} = \tan(\psi) = \mathrm{e_c}\ \tan(\varphi)$$

$$\tan(\varphi) = \frac{\mathbf{T}_{n+1}}{e_c} \quad \text{Thus,} \quad \boldsymbol{\varphi} = \mathrm{atan2}(\ \mathbf{T}_{n+1}\ ,\ e_c\ ) = Sign(z)\,\mathrm{atan2}(\ \mathbf{T}_{n+1}\ ,\ e_c\ ) = \frac{z}{|\ z\ |}\ \mathrm{atan2}(\ \mathbf{T}_{n+1}\ ,\ e_c\ )$$

$$\boxed{\boldsymbol{\varphi = \frac{z}{|\ z\ |}\ atan2(\ T_{n+1}\ ,\ e_c\ )}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }

Geodetic Altitude h Given Geodetic Latitude φ and Reduced Latitude ψ

$$\boxed{p \;= a\;\cos(\psi) + h\;\cos(\varphi)}$$

$$\boxed{|\;z\;| = \mathbf{b}\;\;\sin(\psi) + h\;\;\sin(\varphi)}$$

$$\left.\begin{array}{c} (\;p\;\; - a\;\cos(\psi)\;) \;=\; h\;\cos(\varphi) \\ (\;z\;\; - b\;\sin(\psi)\;) = \;h\;\;\sin(\varphi) \end{array}\right\} \Rightarrow [\;(\;p\;\; - a\;\cos(\psi)\;)\cos(\varphi)\;] \;+\; [\;(\;z\;\; - b\;\sin(\psi)\;)\sin(\varphi)\;] = \;h\;\cos(\varphi\;)\cos(\varphi) + h\;\sin(\varphi\;)\sin(\varphi)$$

$$[\;(\;p\;\; - a\;\cos(\psi)\;)\cos(\varphi)\;] \;+\; [\;(\;z\;\; - b\;\sin(\psi)\;)\sin(\varphi)\;] = \;h\left[\;\cos^2(\varphi) + \sin^2(\varphi)\right] \;=\; h$$

Thus:

$$\boxed{h \;=\; [\;(\;p\;\; - \;a\;\cos(\psi)\;)\cos(\varphi)\;] \;+\; [\;(\;z\;\; - \;\;b\;\sin(\psi)\;)\sin(\varphi)\;]}$$

Compare this result to what was generated using only the **geodetic latitude φ** :

$$h \;= \frac{p}{\cos(\;\varphi\;)} \;-\; \frac{\mathrm{a}}{\sqrt{\;1 - e^2\;\sin^2(\varphi)}}$$

Here use was made of:

$$\boxed{\tan(\;\psi) \;=\; \frac{b}{a}\;\tan(\;\varphi) = \sqrt{\;1\; - \;e^2}\;\tan(\;\varphi) \;= \mathrm{e_c}\;\tan(\varphi) \;=\; (\;1 - f\;)\;\tan(\;\varphi)}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Latitude φ Equation

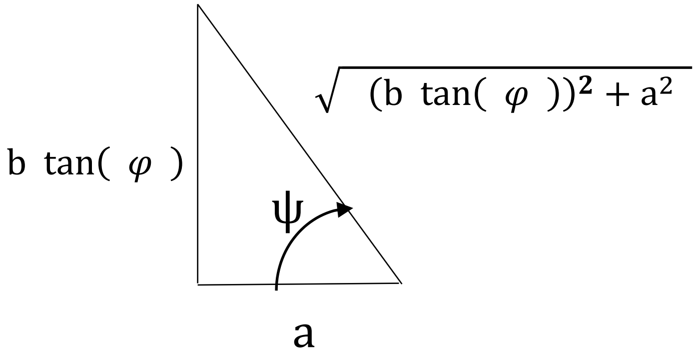
Equation To Be Solved: $\boxed{0 = \mathrm{P}\ \mathbf{T} - \frac{\mathbf{E\ T}}{\sqrt{1+\mathbf{T}^2}} - \mathrm{Z}}$

With: ψ = Reduced Latitude $\mathrm{T} = \tan(\psi)$ $\mathrm{P} = \frac{\mathbf{p}}{\mathrm{a}}$ $\mathrm{E} = \boldsymbol{e^2}$ $\mathrm{Z} = \frac{\sqrt{1-\mathrm{e}^2}}{\mathrm{a}}\ |\ \mathrm{z}\ | = \frac{e_c}{\mathrm{a}}\ |\ \mathrm{z}\ |$

Halley's Third Order Method

Let $\boldsymbol{T_{n+1}}$ be a sufficiently accurate solution generated by Halley's Third Order Iteration Method.

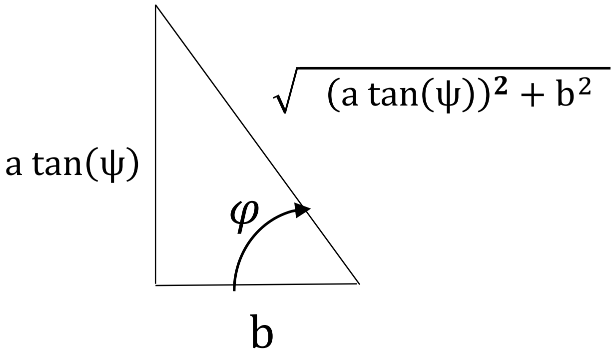
$$\tan(\psi) = \frac{\mathrm{b}}{\mathrm{a}}\ \tan(\varphi)$$



$$\cos(\psi) = \frac{a}{\sqrt{(\mathrm{b}\ \tan(\varphi))^2 + \mathrm{a}^2}}$$

$$\sin(\psi) = \frac{\mathrm{b}\ \tan(\varphi)}{\sqrt{(\mathrm{b}\ \tan(\varphi))^2 + \mathrm{a}^2}}$$

$$\tan(\varphi) = \frac{\mathrm{a}}{\mathrm{b}}\ \tan(\psi)$$



$$\cos(\varphi) = \frac{b}{\sqrt{(\mathrm{a}\tan(\psi))^2 + \mathrm{b}^2}}$$

$$\sin(\varphi) = \frac{\mathrm{a}\tan(\psi)}{\sqrt{(\mathrm{a}\tan(\psi))^2 + \mathrm{b}^2}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Geodetic Altitude h   Given Geodetic Latitude φ

$$\boxed{\boldsymbol{h} = [\,(\,\boldsymbol{p} - \boldsymbol{a}\ \boldsymbol{cos}(\boldsymbol{\psi})\,)\boldsymbol{cos}(\,\varphi\,)\,] + [\,(\,|\,z\,| - \boldsymbol{b}\ \boldsymbol{sin}(\boldsymbol{\psi})\,)\mathbf{sin}(\,\varphi\,)\,]}$$

$$\mathrm{T} = \tan(\,\psi\,) = \frac{\mathrm{b}}{\mathrm{a}}\tan(\,\varphi\,) \implies \tan(\,\varphi\,) = \frac{\mathrm{a}}{\mathrm{b}}\mathrm{T}$$

Let $\boldsymbol{T_{n+1}}$ be a sufficiently accurate solution generated by Halley's Third Order Iteration Method.

$$\tan(\,\varphi\,) = \frac{\mathrm{a}}{\mathrm{b}}\boldsymbol{T_{n+1}}$$

$$\begin{aligned}
\boldsymbol{h} &= [\,(\,\boldsymbol{p} - \boldsymbol{a}\ \mathbf{cos}(\boldsymbol{\psi})\,)\mathbf{cos}(\,\varphi\,)\,] + [\,(\,|\,z\,| - \boldsymbol{b}\ \mathbf{sin}(\boldsymbol{\psi})\,)\mathbf{sin}(\,\varphi\,)\,] \\
&= \left[\left(\boldsymbol{p} - \boldsymbol{a}\frac{a}{\sqrt{(\mathrm{b}\ \tan(\,\varphi\,))^2 + \mathrm{a}^2}}\right)\mathbf{cos}(\,\varphi\,)\right] + \left[\left(|\,z\,| - \boldsymbol{b}\frac{\mathrm{b}\ \tan(\,\varphi\,)}{\sqrt{(\mathrm{b}\ \tan(\,\varphi\,))^2 + \mathrm{a}^2}}\right)\mathbf{sin}(\,\varphi\,)\right]
\end{aligned}$$

$$\sqrt{(\mathrm{b}\ \tan(\,\varphi\,))^2 + \mathrm{a}^2} = \sqrt{\left(\mathrm{b}\frac{\mathrm{a}}{\mathrm{b}}\boldsymbol{T_{n+1}}\right)^2 + \mathrm{a}^2} = \mathrm{a}\sqrt{\boldsymbol{T_{n+1}}^2 + 1}$$

$$\begin{aligned}
\boldsymbol{h} &= \left[\left(\boldsymbol{p} - \boldsymbol{a}\frac{a}{\sqrt{(\mathrm{b}\ \tan(\,\varphi\,))^2 + \mathrm{a}^2}}\right)\mathbf{cos}(\,\varphi\,)\right] + \left[\left(|\,z\,| - \boldsymbol{b}\frac{\mathrm{b}\ \tan(\,\varphi\,)}{\sqrt{(\mathrm{b}\ \tan(\,\varphi\,))^2 + \mathrm{a}^2}}\right)\mathbf{sin}(\,\varphi\,)\right] \\
&= \left[\left(\boldsymbol{p} - \boldsymbol{a}\frac{a}{\mathrm{a}\sqrt{\boldsymbol{T_{n+1}}^2 + 1}}\right)\mathbf{cos}(\,\varphi\,)\right] + \left[\left(|\,z\,| - \boldsymbol{b}\frac{\boldsymbol{b}\frac{\mathrm{a}}{b}\boldsymbol{T_{n+1}}}{\mathrm{a}\sqrt{\boldsymbol{T_{n+1}}^2 + 1}}\right)\mathbf{sin}(\,\varphi\,)\right] \\
&= \left[\left(\boldsymbol{p} - \frac{a}{\sqrt{\boldsymbol{T_{n+1}}^2 + 1}}\right)\mathbf{cos}(\,\varphi\,)\right] + \left[\left(|\,z\,| - \frac{\boldsymbol{b}\ \boldsymbol{T_{n+1}}}{\sqrt{\boldsymbol{T_{n+1}}^2 + 1}}\right)\mathbf{sin}(\,\varphi\,)\right]
\end{aligned}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }

Solution For Transformed Geodetic Altitude h (continued)

$$h = \left[\left(p - \frac{a}{\sqrt{T_{n+1}{}^2 + 1}} \right) \cos(\varphi) \right] + \left[\left(|z| - \frac{b\, T_{n+1}}{\sqrt{T_{n+1}{}^2 + 1}} \right) \sin(\varphi) \right]$$

$$= \left[\left(p - \frac{a}{\sqrt{T_{n+1}{}^2 + 1}} \right) \frac{b}{\sqrt{(a\ T_{n+1})^2 + \mathrm{b}^2}} \right] + \left[\left(|z| - \frac{b\, T_{n+1}}{\sqrt{T_{n+1}{}^2 + 1}} \right) \frac{a\, T_{n+1}}{\sqrt{(a\ T_{n+1})^2 + \mathrm{b}^2}} \right]$$

$$= \left[\left(p - \frac{a}{\sqrt{T_{n+1}{}^2 + 1}} \right) \frac{b}{\mathrm{a} \sqrt{T_{n+1}{}^2 + e_c^2}} \right] + \left[\left(|z| - \frac{b\, T_{n+1}}{\sqrt{T_{n+1}{}^2 + 1}} \right) \frac{a\, T_{n+1}}{\mathrm{a} \sqrt{T_{n+1}{}^2 + e_c^2}} \right]$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ \left[\left(p - \frac{a}{\sqrt{T_{n+1}{}^2 + 1}} \right) \frac{b}{\mathrm{a}} \right] + \left[\left(|z| - \frac{b\, T_{n+1}}{\sqrt{T_{n+1}{}^2 + 1}} \right) T_{n+1} \right] \right\}$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ \left[\left(p - \frac{a}{\sqrt{T_{n+1}{}^2 + 1}} \right) e_c \right] + \left[\left(|z| - \frac{b\, T_{n+1}}{\sqrt{T_{n+1}{}^2 + 1}} \right) T_{n+1} \right] \right\}$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ \left[e_c\ p - \frac{\mathrm{a}\ e_c}{\sqrt{T_{n+1}{}^2 + 1}} \right] + \left[|z|\ T_{n+1} - \frac{b\, T_{n+1}{}^2}{\sqrt{T_{n+1}{}^2 + 1}} \right] \right\}$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ \left[e_c\ p - \frac{\mathrm{a}\ e_c}{\sqrt{T_{n+1}{}^2 + 1}} \right] + \left[|z|\ T_{n+1} - \frac{b\, T_{n+1}{}^2}{\sqrt{T_{n+1}{}^2 + 1}} \right] \right\}$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ e_c\ p + |z|\ T_{n+1} - \frac{\mathrm{a}\ e_c + b\, T_{n+1}{}^2}{\sqrt{T_{n+1}{}^2 + 1}} \right\}$$

$$= \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ e_c\ p + |z|\ T_{n+1} - \frac{b + b\, T_{n+1}{}^2}{\sqrt{T_{n+1}{}^2 + 1}} \right\}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Solution For Transformed Geodetic Altitude h (continued)

$$h = \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ e_c \; p + | \; z \; | \; T_{n+1} \; - b \frac{T_{n+1}{}^2 + 1}{\sqrt{T_{n+1}{}^2 + 1}} \right\} = \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ e_c \; p + | \; z \; | \; T_{n+1} \; - b\sqrt{1 + T_{n+1}{}^2} \right\}$$

Halley's Third Order Method

Let T_{n+1} be a sufficiently accurate solution generated by Halley's Third Order Iteration Method.

Geodetic Latitude: **h**

$$\boxed{\mathrm{h} = \frac{1}{\sqrt{T_{n+1}{}^2 + e_c^2}} \left\{ e_c \; p + | \; \mathrm{z} \; | \; T_{n+1} \; - b\sqrt{1 + T_{n+1}{}^2} \right\}}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Halley's Third Order Method Algorithm Operation Count

$$\left.\begin{array}{l}
\mathbf{T}_{n+1} = \mathbf{T}_n - \dfrac{2\ \mathrm{g}(\mathbf{T}_n)\ \mathrm{g}'(\mathbf{T}_n)}{2\left(\mathrm{g}'(\mathbf{T}_n)\right)^2 - \mathrm{g}''(\mathbf{T}_n)\ \mathrm{g}(\mathbf{T}_n)} \\
\mathrm{g}(\mathbf{T}) = \mathrm{P}\ \mathbf{T} - \dfrac{\mathbf{E}\ \mathrm{T}}{\sqrt{1 + \mathbf{T^2}}} - \mathrm{Z} \\
\mathrm{g}'(\mathbf{T}) = \mathrm{P} - \dfrac{\mathrm{E}}{(1 + \mathbf{T^2})^{\frac{3}{2}}} \\
\mathrm{g}''(\mathbf{T}) = \dfrac{3\,\mathrm{E}\,\mathrm{T}}{(1 + \mathbf{T^2})^{\frac{5}{2}}} \\
\boldsymbol{\varphi} = \dfrac{\boldsymbol{z}}{|\ \boldsymbol{z}\ |}\ \boldsymbol{atan2}(\boldsymbol{T_{n+1}},\ \boldsymbol{e_c}) \\
\mathrm{h} = \dfrac{1}{\sqrt{{T_{n+1}}^2 + e_c^2}} \left\{ e_c\ p + |\ \mathrm{z}\ |\ T_{n+1} - b\sqrt{1 + {T_{n+1}}^2} \right\}
\end{array}\right\} \Rightarrow \begin{cases} 4n\ \ Divisions \\ n + 2\ calls\ to\ sqrt \end{cases}$$

Note(s):

1. Each division operation equals approximately 12 multiplication operations.
2. There are too many divisions.

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h } Halley's Third Order Method Algorithm Separated Into Numerator and Denominator

Note(s):

1. Each division operation equals approximately 12 multiplication operations.
2. There are too many divisions.
3. In order to minimize the number of division operations, we express the variable T in a fractional form, T = S / C, and rewrite Halley's formula, which can be regarded as an iterative formula with respect to the Tangent Function, T , into a pair of iterative formulas in terms of the Sine Function Numerator, S and the Cosine Function Denominator, C.
4. Previous analysis (see Reference [2]) has shown that only one iteration of the Halley's third order method is needed to obtain sufficient accuracy.

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Halley's Third Order Method Algorithm Code Implementation

$$\mathbf{p^2 = x^2 + y^2}, \quad \mathbf{p = \sqrt{x^2 + y^2}}, \quad \mathbf{e^2 = \frac{a^2 - b^2}{b^2}}, \quad \mathbf{e_c = 1 - e^2 = \frac{b}{a}}, \quad \mathbf{p_n = \frac{p}{a}}, \quad \mathbf{E = e^2}, \quad \boldsymbol{z_c = e_c\, S_0}$$

$$\left.\begin{aligned} \mathbf{T_0} &= \frac{|\,z\,|}{e_c\ p} \\ \mathbf{T_0} &= \frac{S_0}{C_0} \end{aligned}\right\} \Rightarrow \left\{ \begin{aligned} S_0 &= \frac{|\,z\,|}{a} \\ C_0 &= e_c\ p_n \end{aligned}\right.$$

Newton Correction Factors:

$$\left\{ \begin{aligned} S_0 &= \frac{|\,z\,|}{a} \\ C_0 &= e_c\ p_n \\ A_0^2 &= S_0^2 + C_0^2 \qquad A_0 = \sqrt{A_0^2} \\ \mathbf{D_0} &= \left(z_c\ A_0^3\right) + \left(E\ S_0^3\right) \\ \mathbf{F_0} &= \left(p_n\ A_0^3\right) - \left(E\ C_0^3\right) \\ \mathbf{B_0} &= \left(1.5\ E^2\right)\ S_0^2\, C_0^2\ p_n\, (A_0 - e_c) \end{aligned}\right.$$

Halley Correction Factors:

$$\left\{ \begin{aligned} \mathbf{S_1} &= (\ \mathbf{D_0}\ \ \mathbf{F_0}\ ) - (\ \mathbf{B_0}\ \ \mathbf{S_0}\ ) \\ \mathbf{C_1} &= (\ \mathbf{F_0}\ \ \mathbf{F_0}\ ) - (\ \mathbf{B_0}\ \ \mathbf{C_0}\ ) \\ \mathbf{a_1} &= \sqrt{(\ e_c^2\ \ S_1^2\ ) + (\ e_c^2\ \ C_1^2\ )} = \sqrt{(\ e_c^2\ \ S_1^2\ ) + C_c^2} \end{aligned}\right.$$

$$C_c = e_c\ \mathbf{C_1}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Halley's Third Order Method Algorithm Code Implementation (continued)

$\mathbf{p^2 = x^2 + y^2}$, $\quad \mathbf{p = \sqrt{x^2 + y^2}}$, $\quad \mathbf{e^2 = \frac{a^2 - b^2}{b^2}}$, $\quad \mathbf{e_c = 1 - e^2 = \frac{b}{a}}$, $\quad \mathbf{p_n = \frac{p}{a}}$, $\quad \mathbf{E = e^2}$, $\quad \boldsymbol{z_c = e_c\, S_0}$

From the Halley's Third Order Method Algorithm:

$$\mathbf{B}_{\boldsymbol{n}} = 1.5\ \mathbf{E\, S_n\, C_n^2}\ [\ (\ \mathbf{P\, S_n - Z\ C_n})\ \mathbf{A_n - E\ S_n\, C_n}\ ]$$

For the initial value n = 0, when:

$$\left.\begin{aligned} \mathbf{T_0} &= \frac{|\ z\ |}{e_c\ p} \\ \mathbf{T_0} &= \frac{\mathbf{S_0}}{C_0} \end{aligned}\right\} \Rightarrow \left\{ \begin{aligned} S_0 &= \frac{|\ z\,|}{a} \\ C_0 &= e_c\ p_n \end{aligned}\right.$$

Then:

$$A_0^2 = S_0^2 + C_0^2, \quad P = \frac{p}{a}, \qquad Z = \frac{\sqrt{1 - e^2}}{a}\ |\ \mathbf{z}\ | = \frac{e_c}{a}\ |\ \mathbf{z}\ |$$

$$\begin{aligned}
\mathbf{B_0} &= 1.5\ \mathbf{E\, S_0\, C_0^2}\ [\ (\ \mathbf{P\, S_0 - Z\ C_0})\ \mathbf{A_0 - E\ S_0\, C_0}\ ] \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ P\ A_0} - 1.5\ \mathbf{E\ S_0\, C_0^3\ Z\ A_0} - 1.5\ \mathbf{E^2\ S_0^2\ C_0^3} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ P\ A_0} - 1.5\ \mathbf{E\ S_0\, C_0^3}\ \frac{\mathbf{e_c}}{\mathbf{a}}\ |\ \mathbf{z}\ |\mathbf{A_0} - 1.5\ \mathbf{E^2\ S_0^2\ C_0^3} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ P\ A_0} - 1.5\ \mathbf{E\ S_0^2\ C_0^3\ e_c\ A_0} - 1.5\ \mathbf{E^2\ S_0^2\ C_0^3} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n\ A_0} - 1.5\ \mathbf{E\ S_0^2\, C_0^2\ C_0\ e_c\ A_0} - 1.5\ \mathbf{E^2\ S_0^2\ C_0^2\ C_0} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n\ A_0} - 1.5\ \mathbf{E\ S_0^2\, C_0^2\ e_c\ p_n\, e_c\ A_0} - 1.5\ \mathbf{E^2\ S_0^2\ C_0^2\ e_c\ p_n} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n} \left\{ \mathbf{A_0} - \mathbf{e_c^2\ A_0} - \mathbf{E\ e_c} \right\} = 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n} \left\{ \mathbf{A_0}\left(\ \mathbf{1} - \mathbf{e_c^2}\right) - \mathbf{E\ e_c} \right\} = 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n} \left\{ \mathbf{A_0\, e^2 - E\ e_c} \right\} \\
&= 1.5\ \mathbf{E\ S_0^2\ C_0^2\ p_n} \left\{ \mathbf{A_0 E - E\ e_c} \right\} = 1.5\ \mathbf{E^2\, S_0^2\ C_0^2\ p_n} \left\{ \mathbf{A_0} - \mathbf{e_c} \right\}
\end{aligned}$$

Thus, just as in the code implementation:

$$\mathbf{B_0} = 1.5\ \mathbf{E^2\ S_0^2\, C_0^2\ p_n}\ (\mathbf{A_0} - \mathbf{e_c})$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Halley's Third Order Method Algorithm Code Implementation (continued)

Solution Value: $\mathbf{T_1} = \dfrac{\boldsymbol{S_1}}{\boldsymbol{C_1}}$ $\mathrm{T} = \tan(\psi) = \mathrm{e_c}\tan(\varphi)$, ψ = Reduced Latitude, φ = Geodetic Latitude

$$\mathbf{T_1} = \tan(\psi) = \mathrm{e_c}\tan(\varphi)$$

$$\tan(\varphi) = \frac{1}{\mathrm{e_c}}\tan(\psi) = \frac{1}{\mathrm{e_c}}\mathbf{T_1}$$

$$\varphi = \mathrm{atan2}(\mathbf{T_1}, \mathrm{e_c})$$

$$\boldsymbol{C_c} = \boldsymbol{e_c}\,\mathbf{C_1}$$

Thus:

$$\boxed{\boldsymbol{\varphi} = \textbf{Geodetic Latitude} = \textbf{atan2}(\mathbf{T_1}, \boldsymbol{e_c}) = \textbf{atan2}(\mathbf{S_1}, \boldsymbol{e_c}\,\mathbf{C_1}) = \textbf{atan2}(\mathbf{S_1}, \mathbf{C}_{\boldsymbol{c}})}$$

Cartesian Coordinates { x y z } Converted To Geodetic Coordinates { φ λ h }
Halley's Third Order Method Algorithm Code Implementation (continued)

Solution Value: $\mathbf{T_1} = \dfrac{\boldsymbol{S_1}}{\boldsymbol{C_1}}$ $\quad \mathrm{T} = \tan(\psi) = \mathrm{e_c}\tan(\varphi)\ ,\quad \psi = \text{Reduced Latitude},\ \ \varphi = \text{Geodetic Latitude}$

$$\boxed{\mathrm{h} = \text{Geodetic Altitude} = \frac{(\boldsymbol{p}\ \boldsymbol{C_c}) + (\mathbf{S_1}\ |\ \boldsymbol{z}\ |) - \boldsymbol{a}\sqrt{(\boldsymbol{e_c^2}\,\boldsymbol{S_1^2}) + \boldsymbol{C_c^2}}}{\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_C^2}}} = \frac{(\boldsymbol{p}\ \boldsymbol{C_c}) + (\mathbf{S_1}\ |\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_1^2}}}{\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_C^2}}}}$$

$$\boldsymbol{C_c} = \boldsymbol{e_c}\ \mathbf{C_1}$$

NOTE(s):

$$\mathrm{h} = \text{Geodetic Altitude} = \frac{(\boldsymbol{p}\ \boldsymbol{C_c}) + (\mathbf{S_1}\ |\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_1^2}}}{\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_C^2}}} = \frac{(\boldsymbol{p}\ \boldsymbol{e_c}\ \mathbf{C_1}) + (\mathbf{S_1}\ |\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_1^2}}}{\sqrt{\boldsymbol{S_1^2} + \boldsymbol{e_c^2}\ \boldsymbol{C_1^2}}}$$

$$= C_1\,\frac{(\boldsymbol{e_c}\ \boldsymbol{p}) + \left(\frac{S_1}{C_1}|\ z\ |\right) - b\sqrt{\left(\frac{S_1}{C_1}\right)^2 + 1}}{C_1\sqrt{\left(\frac{S_1}{C_1}\right)^2 + \boldsymbol{e_c^2}}} = \frac{C_1}{C_1}\,\frac{(\boldsymbol{e_c}\ \boldsymbol{p}) + (\mathbf{T_1}|\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\mathbf{T_1}^2 + 1}}{\sqrt{\mathbf{T_1}^2 + \boldsymbol{e_c^2}}} = \frac{(\boldsymbol{e_c}\ \boldsymbol{p}) + (\mathbf{T_1}|\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\mathbf{T_1}^2 + \mathbf{1}}}{\sqrt{\mathbf{T_1}^2 + \boldsymbol{e_c^2}}}$$

Thus the split iterative method agrees with the regular equation for geodetic altitude:

$$\boxed{\mathbf{h} = \textbf{Geodetic Altitude} = \frac{(\boldsymbol{p}\ \boldsymbol{C_c}) + (\mathbf{S_1}\ |\ \boldsymbol{z}\ |) - \boldsymbol{b}\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_1^2}}}{\sqrt{\boldsymbol{S_1^2} + \boldsymbol{C_C^2}}} = \frac{(\boldsymbol{e_c}\ \boldsymbol{p}) + (|\ \boldsymbol{z}\ |\mathbf{T_1}) - \boldsymbol{b}\sqrt{\mathbf{T_1}^2 + \mathbf{1}}}{\sqrt{\mathbf{T_1}^2 + \boldsymbol{e_c^2}}}}$$

Implementations of an efficient geodetic coordinate transformation

There are five implementations of the geodetic coordinate transformation algorithm.

1. C / C++ implementation
2. Fortran implementation
3. Python implementation
4. Julia implementation
5. Matlab implementation

The implementations involve:

1. C / C++ implementation
   A main program.
   Several functions.
2. Fortran implementation
   A main program.
   Several subroutines.
3. Python implementation
   A main script.
   Several functions.
4. Julia implementation
   A main script.
   Several functions.
5. Matlab implementation
   A main script.
   Several functions.

Implementations of an efficient geodetic coordinate transformation

All testing is accomplished over a trial:

A trial is:

1. Fix the geocentric east longitude at some value (typically 45 [degrees])..
2. Define true geodetic north latitude values from the equator to the north pole (typically every 15 [degrees] of north latitude).
3. At each true geodetic north latitude value, define true geodetic altitude values (typically -10000, 1000000, 2000000 and 3000000 [meters]).
4. Generate true geocentric rectangular x, y and z values based on the true geodetic north latitude, true geodetic altitude and constant geocentric east longitude values.
5. Compute estimated geodetic values based on the true geocentric rectangular x, y, and z values
6. Report accuracy results as the differences between the defined true geodetic values and the estimated geodetic values.

All timing is accomplished by averaging execution time over a large number of trials:

1. Fix a sequence of geocentric east longitude values (every one quarter of a degree from zero degrees east longitude to 359.75 degrees east longitude).
2. Determine the sum of the execution times over all geocentric rectangular to geodetic function calls within each of the specified trials at one of the fixed geocentric east longitude values.
3. Average the sum of execution times over all 1440 trials (one trial at each 0.25 degrees from 0 degrees to 359.75 degrees geocentric east longitude).
4. Divide the average trial execution time by the number of rectangular to geodetic conversion function calls in each trial (4 geodetic altitudes at each of 8 geodetic latitudes so 32 conversion function calls in each trial).

C / C++ Implementation

- **Building C / C++ implementation test program which runs one trial at 45 degrees east geocentric longitude:**

  Within the directory or folder containing the C / C++ source code, use the provided bash shell script to build the program:

```
$ ./buildTestConvertEcefToGeodetic.sh

------------------------------------------------------------------
|
| Building Geodetic Conversion program.
|
------------------------------------------------------------------

------------------------------------------------------------------
|
| Finished building Geodetic Conversion program.
|
|    Program is:-->'./testConvertEcefToGeodetic'
|
------------------------------------------------------------------
```

- **Inspect the prearranged accuracy results of the C / C++ implementation test program for one trial at 45 degrees east longitude:**

  C++ImplementationResults_HPLaptop.txt

- **Inspect the prearranged performance results of the C / C++ implementation test program over all trials:**

  C++ImplementationResultsWithTiming_HPLaptop.txt
  C++ImplementationResultsWithTiming_XeonWorkstation.txt

FORTRAN Implementation

- **Building FORTRAN implementation test program which runs one trial at 45 degrees east geocentric longitude:**

  Within the directory or folder containing the FORTRAN source code, use the provided bash shell script to build the program:

```
$ ./buildTestConvertEcefToGeodetic.sh

-----------------------------------------------------------------
|
| Building Geodetic Conversion program.
|
-----------------------------------------------------------------

-----------------------------------------------------------------
|
| Finished building Geodetic Conversion program.
|
|    Program is:-->'./testConvertEcefToGeodetic'
|
-----------------------------------------------------------------
```

- **Inspect the prearranged accuracy results of the FORTRAN implementation test program for one trial at 45 degrees east longitude:**

  FortranImplementationResults_HPLaptop.txt

- **Inspect the prearranged performance results of the FORTRAN implementation test program over all trials:**

  FortranImplementationResultsWithTiming_HPLaptop.txt
  FortranImplementationResultsWithTiming_XeonWorkstation.txt

Python Implementation

- **Running the Python implementation test program script results in the exercise of the test strategy and generation of a report of the results.**

```
$ python3
Python 3.9.16 (main, Mar  8 2023, 22:47:22)
[GCC 11.3.0] on cygwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import testConvertEcefToGeodetic
```

- **Inspect the prearranged accuracy results of the Python implementation test program for one trial at 45 degrees east longitude:**

  PythonImplementationResults_HPLaptop.txt

- **Inspect the prearranged performance results of the Python implementation test program over all trials:**

  PythonImplementationResultsWithTiming_HPLaptop.txt
  PythonImplementationResultsWithTiming_XeonWorkstation.txt

Julia Implementation

- **Running the Julia implementation test program script results in the exercise of the test strategy and generation of a report of the results.**

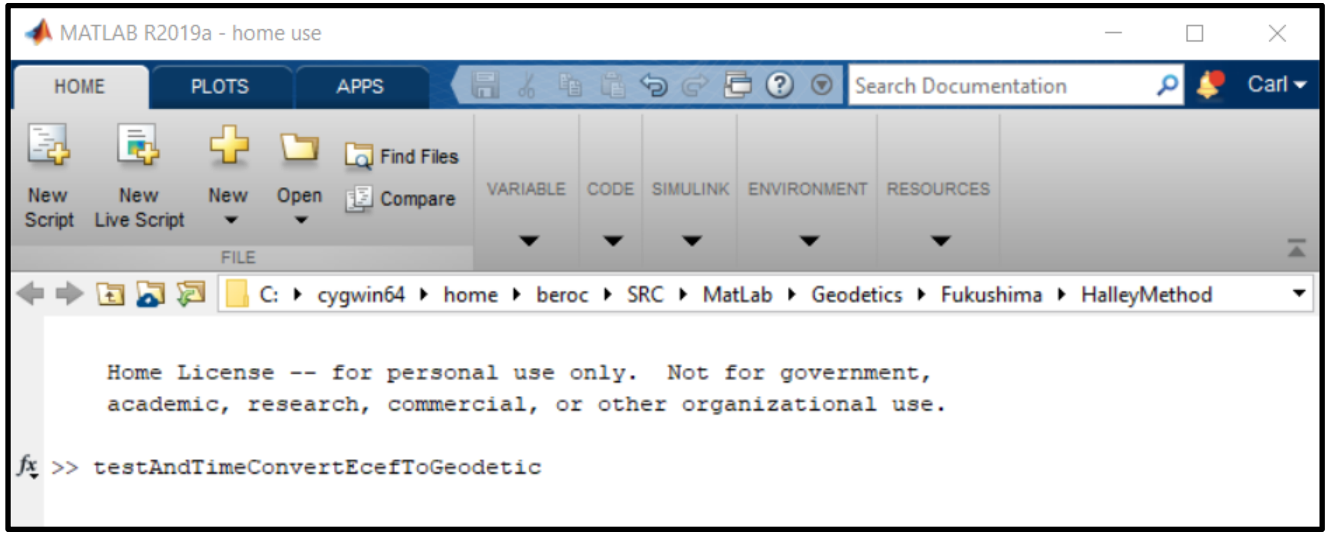
```
$ julia

               _
   _       _ _(_)_     |  Documentation: https://docs.julialang.org
  (_)     | (_) (_)    |
   _ _   _| |_  __ _   |  Type "?" for help, "]?" for Pkg help.
  | | | | | | |/ _` |  |
  | | |_| | | | (_| |  |  Version 1.9.2 (2023-07-05)
 _/ |\__'_|_|_|\__'_|  |  Official https://julialang.org/ release
|__/                   |

julia> include( "./testConvertEcefToGeodetic.jl" )
```

- **Inspect the prearranged accuracy results of the Julia implementation test program for one trial at 45 degrees east longitude:**
  JuliaImplementationResults_HPLaptop.txt

- **Inspect the prearranged performance results of the Julia implementation test program over all trials:**
  JuliaImplementationResultsWithTiming_HPLaptop.txt
  JuliaImplementationResultsWithTiming_XeonWorkstation.txt

Matlab Implementation

- **Running the Matlab implementation test program script results in the exercise of the test strategy and generation of a report of the results.**



- **Inspect the prearranged accuracy results of the Matlab implementation test program for one trial at 45 degrees east longitude:**

  MatlabImplementationResults_HPLaptop.txt

- **Inspect the prearranged performance results of the Matlab implementation test program over all trials:**

  MatlabImplementationResultsWithTiming_HPLaptop.txt
  MatlabImplementationResultsWithTiming_XeonWorkstation.txt

References

[1] "Transformation from Cartesian to geodetic
coordinates accelerated by Halley's method",
Toshio Fukushima,
Journal Of Geodesy (2006),
Volume 79,
Pages 689-693

[2] "Fast transform from geocentric to geodetic coordinates",
Toshio Fukushima,
Journal Of Geodesy (1999),
Volume 73,
Pages 603–610

[3] "Geometric Geodesy, Part A",
"A set of lecture notes which are an introduction to
ellipsoidal geometry related to geodesy.",
R. E. Deakin and M. N. Hunter,
School of Mathematical and Geospatial Sciences,
RMIT University,
Melbourne, Australia,
January 2013
www.mygeodesy.id.au/documents/Geometric%20Geodesy%20A(2013).pdf

[4] 'Various parameterizations of "latitude" equation -
Cartesian to geodetic coordinates transformation',
Marcin Ligas,
Journal of Geodetic Science,
Pages 87 - 94,
2013

[5] "In numerical analysis, Halley's method is a root-finding
algorithm used for functions of one real variable with a
continuous second derivative.",
"The rate of convergence of the iterative Halley's method
is cubic.",
"There exist multidimensional versions of Halley's method.",
wikipedia.org/wiki/Halleys_method

References

[6] “Accurate Algorithm To Transform Geocentric To
Geodetic Coordinates”,
K. M. Borkowski,
Torun Radio Astronomy Observatory,
Nicolaus Copermicus University,
ul. Chopina 12/18, PL–87–100 Torun, Poland
http://www.astro.uni.torun.pl/~kb/Papers/geod/Geod-BG.htm