Australian Government
Department of Defence
Science and Technology

Approximation solutions to the Cartesian to Geodetic coordinate transformation problem

Dr. Hatem Hmam
CEWD, DST Group

Presented at IGNSS 2018, 7-9 Feb, UNSW, Sydney

Cartesian and Geodetic coordinates

Geodetic Coordinates:

Longitude: λ, Latitude: φ and geodetic height, ***h***

Cartesian coordinates: (x, y, z)

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

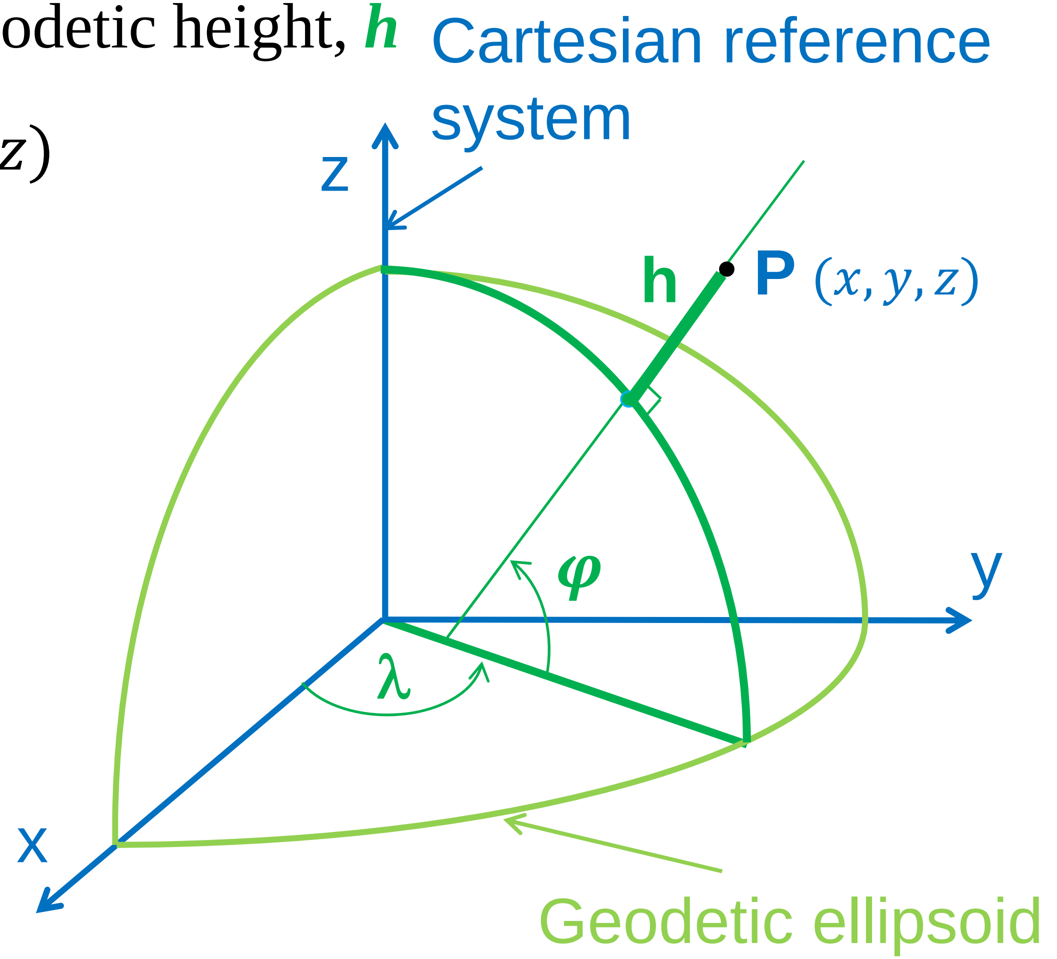
$$y = (N + h)\cos\varphi\sin\lambda$$

$$z = ((1 - e^2)N + h)\sin\varphi$$

$$N = \frac{a}{\sqrt{1 - e^2 sin^2\varphi}}$$

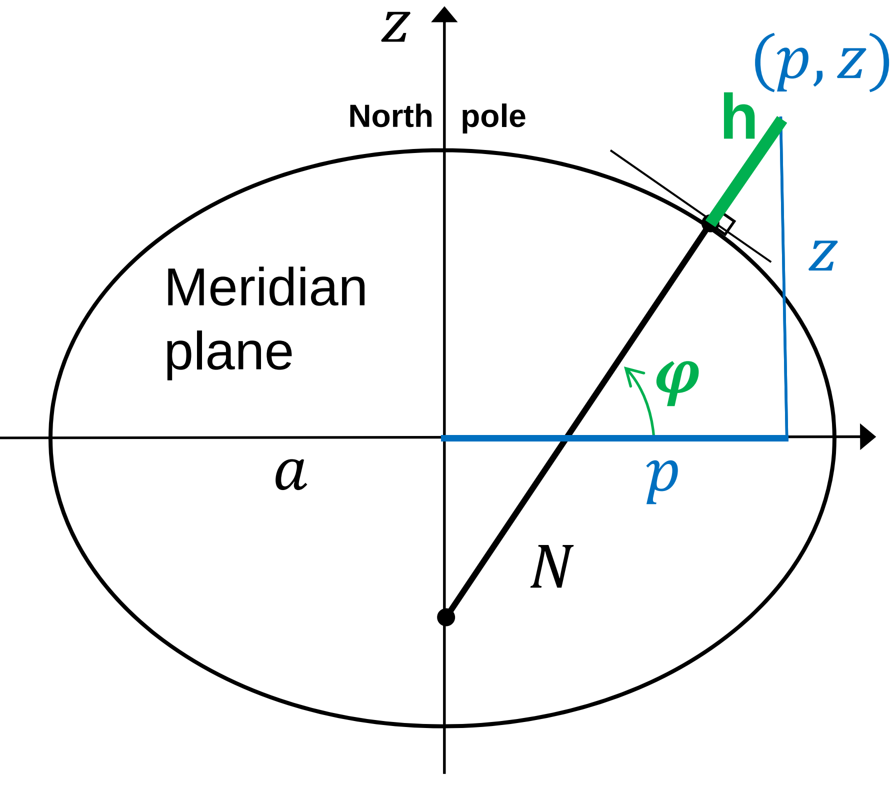
$a = 6378137\,m$ (WGS84)

$e^2 \approx 0.00669437999 \ll 1$



Problem Formulation

Given $\left(p \equiv \sqrt{x^2 + y^2}, z\right)$, find the corresponding Geodetic parameters, (φ, h).



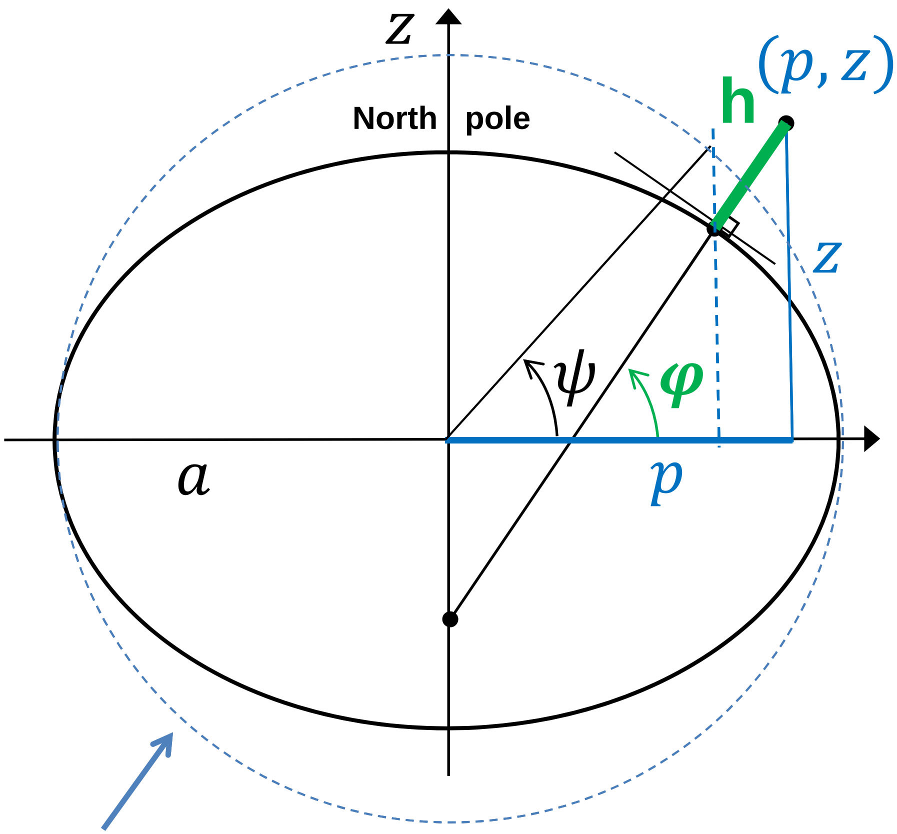
Determine the Geodetic parameters, (φ, h), subject to the two equations

$$p = (N + h)\cos\varphi$$
$$z = ((1 - e^2)N + h)\sin\varphi$$

$$N = \frac{a}{\sqrt{1 - e^2 sin^2\varphi}}$$

$$a = 6378137\,m$$
$$e^2 \approx 0.00669437999$$

Latitude Equation



Auxiliary circle

The angle, ψ, which is known as the reduced latitude, satisfies

$$g(T) \equiv PT - Z - E\frac{T}{\sqrt{1+T^2}} = 0$$

where $T = \tan\psi$, (Fukushima 1999).

The remaining parameters in this equation are $E \equiv e^2$, $P = p/a$, $Z = e_c z/a$ with $e_c = \sqrt{1-E}$

Note: One may encounter numerical problems solving the latitude equation for a Cartesian point on the polar axis. The solution, T, is infinite in this special case.

Fukushima's Solution Method: Halley iteration

(Cubic convergence)

Starting from $$g(T) \equiv PT - Z - E\frac{T}{\sqrt{1+T^2}} = 0$$

Then instead of applying the newton update, $T_{n+1} = T_n - \frac{g(T_n)}{\dot{g}(T_n)}$ (quadratic rate of convergence)

Fukushima applies Halley's method update, to find the approximate the solution of $g(T) = 0$.

$$T_{n+1} = T_n - \frac{g(T_n)}{\dot{g}(T_n) - \ddot{g}(T_n)g(T_n)/(2\dot{g}(T_n))}$$

$$\dot{g}(T_n) = P - \frac{E}{\sqrt{1+T_n^2}} \quad \text{and} \quad \ddot{g}(T_n) = \frac{3ET_n}{\left(\sqrt{1+T_n^2}\right)^5}$$

Initial guess: $T_o = Z/e_c^2 P$

The Geodetic parameters are $\boldsymbol{\varphi_n} = \tan^{-1}\left(\frac{T_n}{e_c}\right)$ and $\boldsymbol{h_n} = \frac{e_c p + zT_n - b\sqrt{1+T_n^2}}{\sqrt{e_c^2 + T_n^2}}$

(One Halley iteration achieves high accuracy conversion, see Table 2 of paper)

Regular Perturbation Method

Given that $E \approx 0.0066943 \ll 1$ (WGS84), regular perturbation theory is used to approximately solve the latitude equation.

Put $T_n = \sum_{i=0}^{n} \alpha_i E^i$. Substitute in the latitude equation and collect all powers of E before setting their coefficients to zero.

With the help of a symbolic processing package such as Matlab Symbolics Toolbox, it turns out that

$$T_n = (Z/P)(1 + \boldsymbol{v}_n^T G_n \boldsymbol{u}_n) = (e_c z/p)(1 + \boldsymbol{v}_n^T G_n \boldsymbol{u}_n)$$

where $\boldsymbol{u}_n = [u^0, u^2, \cdots, u^{2(n-1)}]^T$, $\boldsymbol{v}_n = [v^1, v^2, \cdots, v^n]^T$ and G_n is an $n \times n$ lower triangular square matrix, whose entries are given in Table 1 of paper.

Regular Perturbation Method (continued)

$$T_n = (e_c z/p)(1 + \boldsymbol{v}_n^T G_n \boldsymbol{u}_n)$$

where $\boldsymbol{u}_n = [u^0, u^2, \cdots, u^{2(n-1)}]^T$, $\boldsymbol{v}_n = [v^1, v^2, \cdots, v^n]^T$ and G_n is an $n \times n$ lower triangular square matrix, whose entries are given in the Table below.

$\boldsymbol{G_n}$	u^0	u^2	u^4	u^6	u^8	u^{10}
v	1					
v^2	1	-1				
v^3	1	-7/2	5/2			
v^4	1	-8	15	-8		
v^5	1	-15	427/8	-273/4	231/8	
v^6	1	-25	146	-330	320	-112

$u \equiv \kappa \frac{e_c}{a} z$ and $v \equiv \kappa E$ where $\kappa \equiv a/\sqrt{p^2 + (e_c z)^2}$

With the exception of the first row, the entry sum of each table row is zero.

Regular Perturbation Method (continued)

Put $V_n \equiv pT_n / e_c$, then it follows that

$$V_n = z(1 + \boldsymbol{v}_n^T G_n \boldsymbol{u}_n)$$

The expression on the right hand side is no longer rational

Third Order Perturbation:

$$V_3 = z\big(1 + v\,(1 + (v - w)\,(1 + v - 2.5w)\,)\big)$$

where $v \equiv aE/\sqrt{p^2 + (e_c z)^2}$ and $w \equiv vu^2 = (1 - E)z^2v^3/(aE)^2$.

Fourth and Fifth order perturbation expansions

$$V_4 \equiv z\left(1 + v\left(1 + (v-w)\left(1 + v + v^2 + w\left(8w - 7v - 5/2\right)\right)\right)\right)$$

$$V_5 \equiv z\left(1 + v\left(1 + (v-w)\left(\begin{matrix}(1+v)(1+v^2) \\ +w\left(-\frac{5}{2} + w\left(8 - \frac{231}{8}w\right) + v\left(\frac{315}{8}w - 14v - 7\right)\right)\end{matrix}\right)\right)\right)$$

where $v \equiv aE/\sqrt{p^2 + (e_c z)^2}$ and $w \equiv vu^2 = (1-E)z^2v^3/(aE)^2$.

$\varphi_n = \tan^{-1}(V_n/p)$, if $p \geq |z|$ and $\varphi_n = sign(z)(\pi/2) - \tan^{-1}(p/V_n)$ otherwise

$$h_n = \sqrt{V_n^2 + 1}\left(1 - \frac{a}{\sqrt{e_c^2 V_n^2 + p^2}}\right)$$

Alternative Latitude Formulation

$$ZF - P + E\frac{F}{\sqrt{1+F^2}} = 0 \quad \text{where} \quad F \equiv 1/T$$

$$H_5 \equiv p\left(1 + s\left(1 + (s-w)\left(\begin{matrix}(1+s)\,(1+s^2) \\ +w\left(-\frac{5}{2} + w\left(8 - \frac{231}{8}w\right) + s\left(\frac{315}{8}w - 14s - 7\right)\right)\end{matrix}\right)\right)\right)$$

where $s \equiv -aE/\sqrt{p^2 + (e_c z)^2}$ and $w \equiv st^2 = p^2 s^3/(aE)^2$.

$$\varphi_5 = \tan^{-1}(z/H_5), \qquad h_5 = \sqrt{(z/H_5)^2 + 1}\left(p - \frac{a}{\sqrt{1 + e_c^2 (z/H_5)^2}}\right)$$

Fast approximation algorithm ($n \geq 5$)

$p \geq |z|$:

$$\varphi_n = \tan^{-1}(z/H_n)$$
$$h_n = \frac{1}{E}\left((E-1)p + H_n\right)\sqrt{(z/H_n)^2 + 1}$$

$p < |z|$:

$$\varphi_n = sign(z)\ \pi/2 - \tan^{-1}(p/V_n)$$
$$h_n = \frac{1}{E}\left(|z| + (E-1)|V_n|\right)\sqrt{(p/V_n)^2 + 1}$$

The top formula is more suitable for Cartesian points close or within the equatorial region, whereas the second is more convenient in the polar region

Performance assessment (coordinate conversion accuracy)

Conversion Method	Altitude (km)				
	-10 to 10	10 to 1000	1000 to 20000	20000 to 35000	35000 to 100000
3rd order App.	2.6	2.6	1.66	0.037	0.0094
4th order App.	0.016	0.016	0.009	6.1e-5	8.3e-5
5th order App.	1.2e-4	1.2e-4	5.9e-5	3.3e-5	8.3e-5
Fast Approx.	**0.022**	**0.022**	**0.011**	**0.002**	**0.0047**
Bowring	0.0013	9.5	295	367	453
Fukushima1	3.8e-6	0.004	0.7	0.96	1.32
Fukushima2	0.2	0.2	0.077	0.066	0.135

These are conversion errors in mm

Performance assessment (Expensive arithmetic operations)

Conversion Method	Expensive Arithmetic Operations		
	Division	Square root	Arc Tangent
3rd order Approx.	3	4	1
4th order Approx.	3	4	1
5th order Approx.	3	4	1
Fast Approx.	2	3	1
Bowring (1 iter.)	4	4	1
Fukushima1 (1 iter.)	4	4	1
Fukushima2 (1 iter.)	2	4	1

A Fortran program was written to compare the algorithm runtimes on a desktop with the processor/memory specifications: Intel i7-2600 3.4GHz, RAM 8.00 GB. The obtained runtimes are approximately **61.3, 61.4, 65.5, 60.7, 74.3, 79.0, and 72.2 ns**

Questions ???