A Comparative Analysis of the Performance of Iterative and Non-iterative Solutions to the Cartesian to Geodetic Coordinate Transformation

Hok Sum Fok and H. Bâki Iz

Department of Land Surveying and Geo-informatics, The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

Abstract:

We examined four non-iterative and four iterative methods for the computation of ellipsoidal latitude and height from the Cartesian coordinates using a performance index, defined by the ratio of the each method's CPU time to a reference method's CPU time; hence the inferences about the performance of the methods are less dependent on the computing platform. We find that the iterative methods are faster than the non-iterative methods by a factor of three. Almost all of the iterative methods quickly converge to sub millimeter level right after two iterations, an accuracy which exceeds the requirements of any practical application.

1 Introduction

Various regional and global surveying practices, such as GPS, mapping applications, navigation frequently use Cartesian to geodetic coordinate transformations and visa versa. The transformation from Geodetic to Cartesian coordinates is well known, simple and straightforward whereas the inverse computation, namely computation of the geodetic latitude from Cartesian coordinates is used to be iterative (Heiskanen and Moritz 1967). With the proliferation of symbolic calculation using desk top computers, non-iterative closed form solutions are now available. Although the accuracy of the closed form solutions is only limited by the precision arithmetic, approximate iterative solutions which are simpler are desirable if is their convergence speed is faster than the computational speed of non-iterative methods. For instance, real-time display of sub satellite points of long arcs of an artificial satellite involves several hundred thousands of transformations in an earth fixed coordinate system. The computation speed also matters in real time applications that involve similar transformations.

This study revisits the comparison of the Cartesian to geodetic coordinate transformation problem to assess the performance of iterative and newly developed non-iterative methods in terms of their accuracy and their execution speed.

We first list a compendium of transformation equations that are iterative and non-iterative in nature, and then compare their performances as a function of ellipsoidal latitudes and heights of regularly sampled points above a reference ellipsoid using a performance index.

2 Iterative Solutions

Iterative methods involve:

a) Trigonometric functions such as those from Heiskanen and Moritz (1967), Bowring (1976), Fukushima (1999), Jones (2002), Seemkooei (2002)
b) Iterative vector methods such as those proposed by Lin and Wang (1995), Guo (2001), and Pollard (2002). Among those we choose the following four methods for comparison.

2.1 Heiskanen and Moritz's Method (1967)

This method iteratively solves for the geodetic latitude φ. It is well known and frequently used due to its simplicity. Computations are as follows.

Journal of Geospatial Engineering, Vol. 5, No.2 (December 2003), pp.61-74.

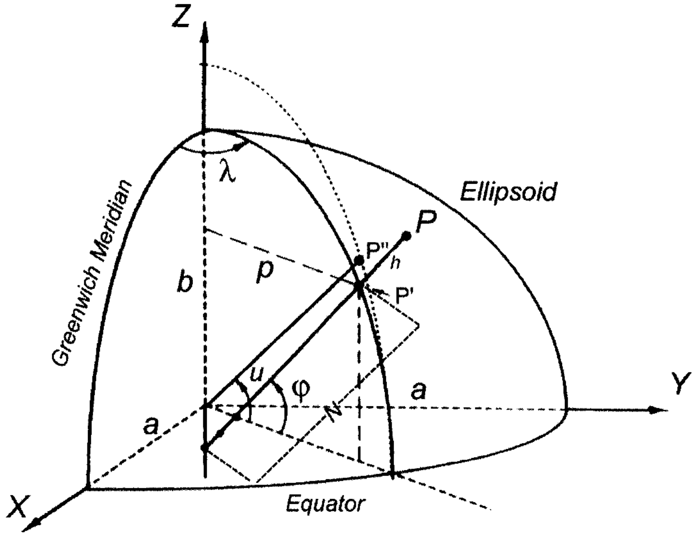

Figure 1. Geodetic and Cartesian Coordinates.

From the given Cartesian coordinates X and Y of a point the radius of the parallel p (Figure 1. is computed using the following expression

$$p = \sqrt{X^2 + Y^2} = (N + h)\cos\varphi \qquad (1)$$

where a is the semi major axis of the reference ellipsoid, h is the height above the ellipsoid and N is the radius of curvature in the prime vertical which is given by

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

where e is the first eccentricity. Equation (1) can be rearranged as

$$h = \frac{p}{\cos\varphi} - N \qquad (3)$$

Considering the first eccentricity $e^2 = (a^2 - b^2)/a^2$ we get,

$$Z = (N + h - e^2 N)\sin\varphi \, . \qquad (4)$$

Dividing the equation (4) by (1) yields to

$$\tan\varphi = \frac{Z}{p}(1 - e^2 \frac{N}{N+h})^{-1} \, . \qquad (5)$$

The iterative procedure is initiated by computing an approximate value φ_0 using

$$\tan\varphi_0 = \frac{Z}{p}(1 - e^2)^{-1} \qquad (6)$$

then continues calculating the approximate values N_0 and h_0

$$N_0 = \frac{a^2}{\sqrt{a^2 \cos^2 \varphi_0 + b^2 \sin^2 \varphi_0}}$$

$$h_0 = \frac{p}{\cos \varphi_0} - N_0 \tag{7}$$

Once the starting approximate values are available for φ and h the following equations are repeatedly computed until they converge

$$\varphi_i = \tan^{-1} \frac{Z}{p} (1 - e^2 \frac{N_i}{N_i + h_i})^{-1}$$

$$N_i = \frac{a^2}{\sqrt{a^2 \cos^2 \varphi_i + b^2 \sin^2 \varphi_i}}$$

$$h_i = \frac{p}{\cos \varphi_i} - N_i \tag{8}$$

2.2 Bowring's Method (1976)

This method is one of the oldest, frequently used and well known approach for its fast convergence. It uses Newton's iterative technique and converges usually after the second or third iteration. The order of computational steps is as follows.

a) Make an initial guess for reduced (parametric) latitude u using

$$u_0 = \tan^{-1}\left(\frac{a}{b}\frac{Z}{p}\right) \tag{9}$$

b) Evaluate φ using

$$\varphi = \tan^{-1} \frac{b}{a} \frac{Z + \varepsilon b \sin^3 u}{X - e^2 a \cos^3 u} \tag{10}$$

where the first eccentricity of the reference ellipsoid is defined as $e^2 = (a^2 - b^2)/a^2$, ε is the square of the second eccentricity of the ellipsoid given by $\varepsilon = (a^2 - b^2)/b^2$.

c) Update u using

$$u = \tan^{-1}\left(\frac{b}{a}\tan \varphi\right) \tag{11}$$

Computation continues repeating steps (ii) through (iii) until φ converges.

2.3 Lin and Wang's Method (1995)

This method also uses Newton's iterative technique to solve for a scalar multiplier m and calculate φ and h from (16) and (17)

(18) in order. The sequence of the necessary computations is as follows.

Given the Cartesian coordinates *X, Y, Z,* for a point related to an ellipsoid, m_0 is calculated using the following expression

$$m_0 = \frac{ab(a^2Z^2 + b^2p^2)^{1/2} - a^2b^2(a^2Z^2 + b^2p^2)}{2(a^4Z^2 + b^4p^2)} \tag{12}$$

Improved values of is m *is* obtained iteratively using

$$m_{n+1} = m_n - \frac{f(m_n)}{f'(m_n)} \tag{13}$$

where the function and its first derivative are

$$f(m_n) = \frac{p^2}{(a + \frac{2m_n}{a})^2} + \frac{Z^2}{(b + \frac{2m_n}{b})^2} - 1 \tag{14}$$

$$f'(m_n) = -4\left\{ \frac{p^2}{a(a + \frac{2m_n}{a})^3} + \frac{Z^2}{b(b + \frac{2m_n}{b})^3} \right\} \tag{15}$$

After calculating m, p_E and Z_E are given by the following expressions

$$p_E = \frac{p}{1 + \frac{2m}{a^2}} \quad Z_E = \frac{Z}{1 + \frac{2m}{b^2}} \tag{16}$$

and the ellipsoidal latitude φ and height h are obtained from

$$\tan\varphi = \frac{a^2 Z_E}{b^2 p_E} \tag{17}$$

$$h = \pm\sqrt{(p - p_E)^2 + (Z - Z_E)^2} \tag{18}$$

h is negative if $(p_E + |Z|) < (p_E + |Z_E|)$.

2.4 Jones' Method (2002)

This method is also based on Newton's iterative solution to solve for the parametric latitude u. The order of necessary equations is given below.

To calculate the starting value u_0, for u three regions are considered and different formulae is applied for each region as follows.

For the northern hemisphere:

Region (1): If the point in question lies on or outside the ellipse defined by the following expression

$$\frac{p^2}{a^2} + \frac{Z^2}{b^2} \geq 1 \tag{19}$$

we use

$$u_0 = \tan^{-1}\left(\frac{Z}{p\sqrt{1 - e^2}}\right) \tag{20}$$

Region (2): If the point in question lies inside the ellipse with high latitude and passes through the 2 vertices of the evolute where

$$\frac{p^2}{a^2}+\frac{Z^2}{b^2}<1 \text{ and } p \le p^* + \frac{Z}{\sqrt{1-e^2}} \tag{21}$$

We use

$$u_0 = \tan^{-1}\left(\frac{Z\sqrt{1-e^2}+p^*}{p}\right) \tag{22}$$

Region (3): If the point in question lies inside the ellipse with low latitude and passes through the 2 vertices of the evolute where

$$\frac{p^2}{a^2}+\frac{Z^2}{b^2}<1 \text{ and } p \ge p^* + \frac{Z}{\sqrt{1-e^2}} \tag{23}$$

we use

$$u_0 = \tan^{-1}\left(\frac{Z\sqrt{1-e^2}}{p-p^*}\right) \tag{24}$$

In the above expressions $p^* = ae^2 = (a^2-b^2)/a$

A *reduced-latitude characteristic function* is defined to calculate the reduced-latitude (i.e. u)

$$f(u) = \tan^{-1}\left(\frac{bZ}{ap}+\frac{p^*}{p}\sin u\right) \tag{25}$$

If (p, z) is fixed in space and let T be a proper reduced latitude i.e. $\operatorname{sgn} T = \operatorname{sgn} z$ and $T = f(T)$, T will be a root of $F(u) = f(u) - u$

$$F(u) = \tan^{-1}\left(\frac{bZ}{ap}+\frac{p^*}{p}\sin u\right) - u \tag{26}$$

Then, by using the Newton's iteration, i.e.

$$u_{n+1} = u_n - \frac{F(u_n)}{F'(u_n)} \tag{27}$$

the proper reduced latitude is obtained. Using this information the ellipsoidal latitude φ is calculated from the reduced latitude using the following well-known expression

$$\varphi = \tan^{-1}\left(\frac{a}{b}\tan u\right). \tag{28}$$

The ellipsoidal height h is obtained from

$$h = h_1\cos^2(\varphi) + h_2\sin^2(\varphi) \tag{29}$$

where

$$h_1 \equiv p/\cos(\varphi) - N \text{ and } h_2 \equiv z/\sin(\varphi) - N(1-e^2) \tag{30}$$

or

$$h \equiv p\cos\varphi + Z\sin\varphi - a\sqrt{(1-e^2\sin^2\varphi)} \tag{31}$$

3 Non-iterative Solutions

There are a multitude of non-iterative solutions now available for the transformation including the ones by Paul (1973), Vicenty (in Groten, 1979), Ozone (1985), Bordowski (1989), You (2000), and Vermeille (2002). We choose the following methods for testing. Note that You's method is not a closed from expression but only a non-iterative approximate solution based on series expansion.

3.1 Paul's Method (1973)

This method is a closed formula to solve for the geodetic latitude φ after several intermediate variables are calculated. The order of necessary equations is given below.

From (1) and (5) we write

$$p\tan\varphi - Z = e^2 N \sin\varphi \tag{32}$$

Substituting the expression for N for the radius of curvature of the prime vertical into the above expression we get

$$p\tan\varphi - Z = \frac{ae^2\sin\varphi}{\sqrt{1-e^2\sin^2\varphi}} \tag{33}$$

By dividing the numerator and denominator of the right-hand side by $\cos\varphi$ and squaring the whole equation yields

$$p^2\tan^4\varphi - 2pZ\tan^3\varphi + (Z^2 + \frac{p^2 - a^2e^4}{1-e^2})\tan^2\varphi - \frac{2pZ}{1-e^2}\tan\varphi + \frac{Z^2}{1-e^2} = 0 \tag{34}$$

This is a quartic (biquadratic) equation in $\tan\varphi$, in which the values of all the coefficients are known. Its closed form solution gives the following expression to compute the ellipsoidal latitude φ

$$\varphi = \tan^{-1}\left\{\left(\frac{Z}{2} + \sqrt{\tau} + \sqrt{-\frac{\beta}{2} + \frac{Z^2}{4} - \tau + \frac{\alpha Z}{4\sqrt{\tau}}}\right)\Big/ p\right\} \tag{35}$$

where

$$\tau \equiv \frac{z^2+\beta}{12}\left(\sqrt[3]{q+\sqrt{q^2-1}} + \sqrt[3]{q-\sqrt{q^2-1}}\right) - \frac{\beta}{6} + \frac{Z^2}{12} \tag{36}$$

$$q \equiv 1 + \frac{27Z^2(\alpha^2-\beta^2)}{2(Z^2+\beta)^3} \tag{37}$$

$$\alpha \equiv \frac{p^2 + a^2e^4}{1-e^2} \tag{38}$$

$$\beta \equiv \frac{p^2 - a^2e^4}{1-e^2} \tag{39}$$

3.2 Ozone's Method (1985)

This method is a closed formula to solve for φ after several intermediate variables are calculated. The order of necessary equations is listed below.

The ellipsoidal latitude φ can be computed by solving for u using the following quartic equation

$$u^4 - 4Mu^3 - 4Nu - 1 = 0 \tag{40}$$

which is given by

$$\varphi = \tan^{-1}\left(\frac{2au}{b(u^2-1)}\right) \tag{41}$$

where

$$M \equiv \frac{ap - (a^2 - b^2)}{2bZ} \tag{42}$$

$$N \equiv \frac{ap + (a^2 - b^2)}{2bZ} \tag{43}$$

$$V \equiv 4NM + 1 \tag{44}$$

$$W \equiv 2(N^2 - M^2) \tag{45}$$

$$I \equiv \left(\sqrt{\left(\frac{V}{3}\right)^3 + \left(\frac{W}{2}\right)^2} + \frac{W}{2}\right)^{1/3} - \left(\sqrt{\left(\frac{V}{3}\right)^3 + \left(\frac{W}{2}\right)^2} - \frac{W}{2}\right)^{1/3} \tag{46}$$

$$J \equiv \sqrt{2I + 4M^2} \tag{47}$$

$$K \equiv \frac{2(N - MI)}{J} \tag{48}$$

$$G \equiv (2M + J)^2 - 4(I - K) \tag{49}$$

$$u \equiv \frac{2M + J + \sqrt{G}}{2} \tag{50}$$

The quadrant of the geodetic latitude φ is determined by inspecting the Cartesian coordinate Z.

3.3 Borkowski's Method (1989)

This method is a closed form to solve for φ after several intermediate variables are calculated. Given p and z, the problem of finding φ, and h can be reduced to solve the following equation

$$2\sin(u - \Omega) - c\sin 2u = 0 \tag{51}$$

where $\Omega \equiv \tan^{-1}[(bz)/(ar)]$, $c \equiv (a^2 - b^2)/[(ar)^2 + (bz)^2]^{1/2}$ and u is the reduced latitude. The geodetic latitude φ can be determined by equation (27).

The above expression, when expressed in $t \equiv \tan(\pi/4 - u/2)$, gives the following quartic (or biquadratic)

$$t^4 + 2Et^3 + 2Ft - 1 = 0 \tag{52}$$

where

$$E = [bz - (a^2 - b^2)]/(ar) \tag{53}$$

$$F = [bz + (a^2 - b^2)]/(ar) \tag{54}$$

By applying the Ferrari's solution (Korn and Korn, 1968) to the quartic defined above we obtain a solution for t

$$t = \pm\sqrt{G^2 + \frac{F - vG}{2G - E}} - G \tag{55}$$

where

$$G \equiv [\pm(E^2 + v)^{1/2} + E]/2 \tag{56}$$

$$v \equiv (D^{1/2} - Q)^{1/3} - (D^{1/2} + Q)^{1/3} \tag{57}$$

$$D \equiv P^3 + Q^2 \tag{58}$$

$$P \equiv \frac{4}{3}(EF + 1) \tag{59}$$

$$Q \equiv 2(E^2 - F^2) \tag{58}$$

The equation (55) has four solutions; they correspond to the four combinations of $\pm$ sign in (55) and value of G. For $D > 0$ all four solutions are real while for $E > 0$ two solutions are real and the two remaining solutions are complex. To obtain a programmable algorithm yielding to a unique solution we just omit the double signs in equation (55) and (56) to leave only the positive square roots (provided that $a > b$ and $\varphi > 0$.)

Distinct geodetic coordinates φ, h are obtained from the following equations

$$\varphi = \tan^{-1}[a(1 - t^2)/(2bt)] \tag{60}$$

$$h = (r - at)\cos\varphi + (z - b)\sin\varphi \tag{61}$$

3.4 Vermeille's Method (2002)

This method is a closed form to solve for φ with several intermediary variables. It introduces the strictly positive coefficient for algebraic computation of the values of h and φ without ambiguity. Then, by a series of derivations using the basic properties of ellipse and algebraic relationships a series of equations for calculating φ, h, are derived. The order of necessary equations is given below.

Using the given Cartesian coordinates the following intermediary parameters are calculated

$$p = \frac{X^2 + Y^2}{a^2} \tag{62}$$

$$q = \frac{1 - e^2}{a^2} Z^2 \tag{63}$$

$$r = \frac{p + q - e^4}{6} \tag{64}$$

$$s = e^4 \frac{pq}{4r^3} \tag{65}$$

$$t = \sqrt[3]{1 + s + \sqrt{s(2 + s)}} \tag{66}$$

$$u = r(1 + t + \frac{1}{t}) \tag{67}$$

$$v = \sqrt{u^2 + e^4 q} \tag{68}$$

$$w = e^2 \frac{u + v - q}{2v} \tag{69}$$

The above values enable the computation of the following variables *k* and *D*

$$k = \sqrt{u + v + w^2} - w \tag{70}$$

$$D = \frac{k\sqrt{X^2 + Y^2}}{k + e^2} \tag{71}$$

which are used in computing geodetic coordinates (φ, λ, h) using the following expressions

$$\lambda = 2\tan^{-1}\left(\frac{Y}{X + \sqrt{X^2 + Y^2}}\right) \tag{72}$$

$$\varphi = 2\tan^{-1}\left(\frac{Z}{D + \sqrt{D^2 + Z^2}}\right) \tag{72}$$

$$h = \frac{k + e^2 - 1}{k}\sqrt{D^2 + Z^2} \tag{73}$$

4 Comparison criterion and Design for the Performance Testing

In assessing the performance of a transformation method over the other we use accuracy and speed of computation (rate of convergence for iterative methods) as criterions.

Since in all cases the computation of geodetic longitude is based on a closed form we will only consider geodetic latitude and height errors which are defined as

$$\Delta\varphi = |\varphi_j - \varphi_i| \tag{734a}$$

$$\Delta h = | h_j - h_i | \tag{74b}$$

where (φ_i, h_i) are the assigned *known* geodetic coordinates and (φ_j, h_j) are the geodetic coordinates calculated from one of the transformation methods using the Cartesian counterparts obtained using the well known geodetic to Cartesian transformation.

Geodetic longitude is fixed in all computations at $\lambda = 114^0$ and calculated using one of the equations below

$$\lambda = \tan^{-1}\frac{Y}{X} \tag{75}$$

$$\lambda = 2\tan^{-1}\left(\frac{Y}{X + \sqrt{X^2 + Y^2}}\right) \tag{76}$$

In order to evaluate accuracy and execution speed of each method, each method is coded using Matlab version 6.1 on a PC with 550MHz Pentium processor. For each algorithm, the CPU time required to perform the Cartesian to geodetic transformations for points is recorded. The CPU time for preparing the constants, such as $p^* = ae^2$ and $\varepsilon = (a^2 - b^2)/b^2$, is excluded from the measurements but the time used for computing the starter value is incorporated in the total computation time.

Each method is tested for a number of regularly sampled points at different latitudes at 5^0 intervals over the range $1^0 \leq \varphi \leq 89^0$ (because Paul's and Ozone's methods fail at $Z = 0$ we started the iteration at 1^0) for all the ellipsoidal height $0 \leq h \leq 100000$ m at 100 m intervals and the total CPU time is reported for each latitude. Note that this problem can be avoided also by checking the Z coordinate whether it is equal to zero before the computations. In the case of small Z it is possible to modify the algorithms to prevent the loss of significant digits which is not attempted in this study. A few sample runs with both methods show that both methods give reasonable results despite the small Z values as long as the compiler continues to operate with the extreme intermediary results.

Similarly we calculate a number of regularly sampled points at different heights $0 \leq h \leq 100000$ m at 1000 m intervals for all the latitudes at 0.1° intervals. And again the total CPU time is reported for each latitude.

Since processing speed changes rapidly over time (Moor's Law) we use the following quotient to define a *performance index, PI,* which is less dependent on computer processor technology

$$PI := \frac{\text{Processing Speed of a Method}}{\text{Processing Speed of the Reference Method}} \tag{77}$$

As for the parameters of reference ellipsoid, the WGS 84 system is adopted where a = 6378137 m, and inverse flattening = 298.257223563. Double precision arithmetic, allowing 16 significant digits, is used throughout the calculations.

Because the current geodetic coordinate accuracies can reach to millimeter level for some observational technologies, we adopted convergence criterions of less than 0.5 mm for the geodetic height and 0.5×10^{-5} arc second for the geodetic latitude (corresponding approximately to 0.2 mm on the surface of the ellipsoid) and used them for convergence criterions to stop the iterations.

For the test computations the working Cartesian coordinates (X_i, Y_i, Z_i) are obtained from $(\varphi_i, \lambda_i, h_i)$ using the following well known parametric transformation formulas

$$X = (N + h)\cos\varphi\cos\lambda \tag{78}$$

$$Y = (N + h)\cos\varphi\sin\lambda \tag{79}$$

$$Z = (\frac{b^2}{a^2}N + h)\sin\varphi \tag{80}$$

We then compute the geodetic latitude φ_j of each point using the relevant equations for each method (i.e. both iterative and non-iterative methods).

The ellipsoidal height h_j is determined using the following relationship as it is more stable when φ approaches zero (Borkowski 1989) unless specified by a particular method (e.g. Heiskanen and Moritz 1967, Lin and Wang 1995, Jones 2002, Vermeille 2002).

$$h = (p - a\cos u)\cos\varphi + (z - b\sin u)\sin\varphi \qquad (81)$$

Comparison of calculated φ_j and h_j with their known counterparts gives the magnitude of the geodetic latitude error $\Delta\varphi$ and the ellipsoidal height error Δh at regularly sampled latitudes and heights all of which are compared against the convergence criterion stated above for the iterative methods.

5 Numerical Results

We examined the performance of the iterative and non-iterative methods as a function of ellipsoidal latitude and height.

5.1 Performance as a Function of Ellipsoidal Latitude

In general, iterative methods provide an acceptable accuracy for latitude dependence error after the second iteration, except for Heiskanen & Moritz's method. All the other iterative methods make use of the Newton's iterative method which provides quadratic convergence for convergence. Despite that Heiskanen and Moritz method is the fastest method among all the methods considered (Figure 2). Overall all the iterative methods we used perform better than the non iterative methods.

The performance of the iterative methods is not significantly different from each other. The performance index oscillates about 1.5 for the Jones and Lin and Wang method. Figure 2 also shows that there is no significant performance dependency to the station position as a function of latitude.

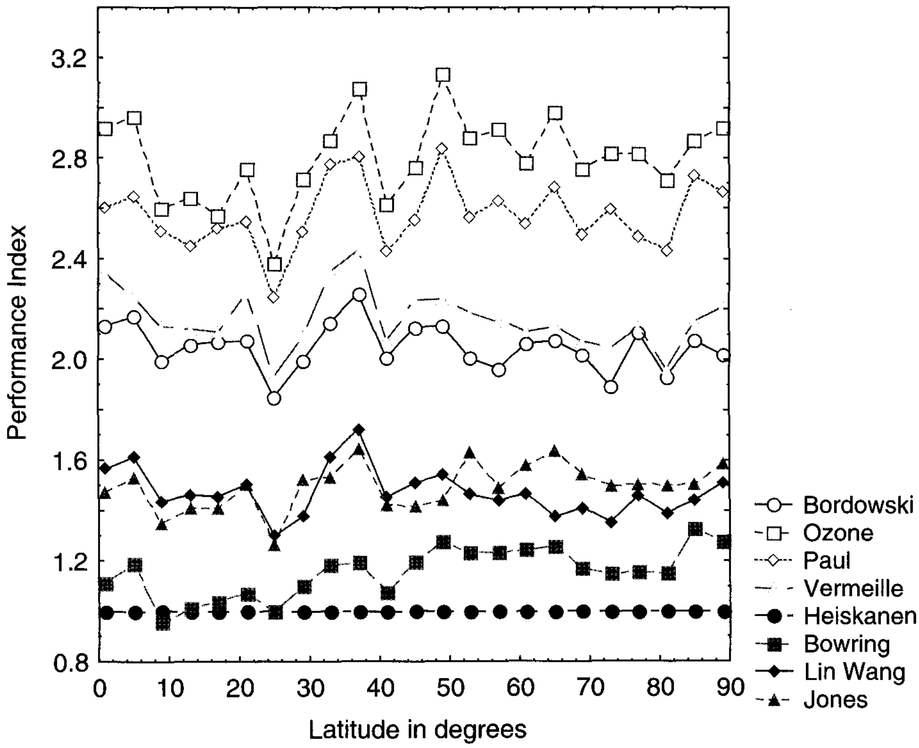

Figure 2. Performance of non-iterative and iterative methods with respect to Heiskanen and Moritz Method at different latitudes.

The convergence speed of the Bowring method is very close to the Heiskanen and Moritz and if only the latitude convergence is used as a criterion Bowring method performs better than the Heiskanen and Moritz method. In the past different authors have used different approaches on how to assess the execution speed for their methods and obtained somewhat different results in speed than the ones we obtained (e.g. Seemkooe 2002, Gerdan and Deakin 1999, Lin and Wang

1995, Laskowski 1991) . Nevertheless the performance index we used seems to be more suited for comparison but it may not be the best if the impact of the technology on the calculations for different methods is not the same. In any case, the use of convergence in ellipsoidal height together with convergence in ellipsoidal latitude is more justifiable than the convergence in latitude only.

In specific, we observed that the starter value (i.e. approximate value for latitude) for Jones' method is bad to use without iteration. Bowring's method always provides acceptable accuracy for both approximate value and the first iteration, while Ling and Wang's method provides sufficient accuracy after the first iteration.

As for the non-iterative methods, all of them provide exact answers in both the ellipsoidal latitude φ and height h, hence only affected by the precision arithmetic. It has been shown that Paul's and Ozone's method are numerically instable when approaching to 0°. As reported by Gerdan and Deakin 1999 that Paul's and Ozone's methods are not used when $Z = 0$ or very small, in which cases they either fail or give unreliable result.

Figure 2 also shows that the non-iterative methods are distinctly slower that the iterative methods as a group. Their performance indices do not exhibit dependency on the ellipsoidal latitude.

5.2 Performance as a Function of Ellipsoidal Height

For the comparison between iterative and non-iterative methods on height dependence, we observed that the accuracy of iterative methods attained after 2 iterations is acceptable for all practical applications. Similar to our previous conclusion in the case of latitude dependency, all the iterative methods are faster than the non-iterative methods (Figure 3).

Seemkooei (2002) reports that Lin and Wang and Heiskanen and Moritz methods' CPU time increase with increasing height. However, we found that the increase is not very significant up to 100 km in our study (Figure 3).

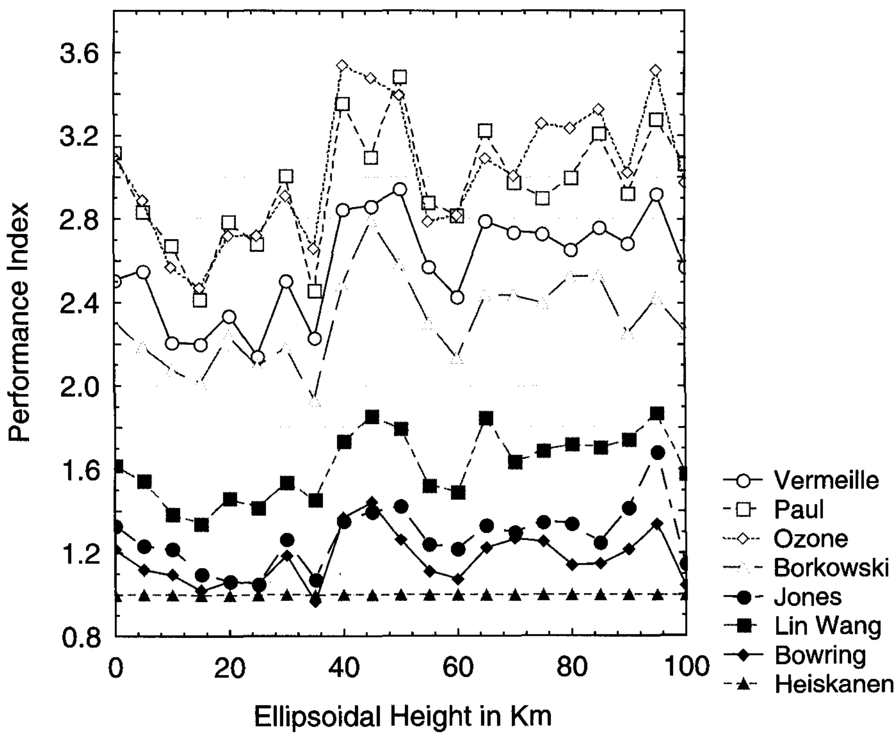

Figure 3. Performance of non-iterative and iterative methods with respect to Heiskanen and Moritz Method at different ellipsoidal heights.

Iterative methods provide an acceptable accuracy for height dependence error after the second iteration, which is the same, as discussed in the last section for the latitude dependence error.

The slowest method amongst eight methods described in this study is the Ozone's method which is due to a number of substitutions for calculating new variables which consume significant amount of CPU time.

6 Conclusion

Overall we find that the iterative methods are faster than the non-iterative methods. Iterative methods quickly converge to millimeter level right after two iterations, an accuracy which exceeds the requirements of any practical application. Among all the approaches Heiskanen and Moritz method is the fastest one when the solution is required to converge both in geodetic latitude and ellipsoidal height. Bowring method is the fastest if only convergence in the latitude is required. Both methods are equally suited for any requirement. Iterative methods are faster than their non-iterative methods sometimes by a factor of three.

Among the non-iterative methods, Ozone's approach is the slowest one and may require special attention when the geodetic latitude is close to zero for numerical stability for certain compilers. As far as the complexity of the algorithms is concerned, all the methods are simple enough for easy coding.

Acknowledgement

This manuscript is a by-product of a more extensive study on the transformation methods by the first author's BSc dissertation in the Department of Land Surveying and Geo-informatics at the Hong Kong Polytechnic University. The study was partially supported by a grant from the RGC allocation project no. G-T423, The Hong Kong Polytechnic University, China.

References

Borkowski K.M., 1987, Transformation of geocentric to geodetic coordinates without approximations, *Astronomical Space Sciences,* 139:1–4.

Borkowski K.M., 1989, Accurate algorithms to transform geocentric to geodetic coordinates, *Bulletin Geodesique,* 63:50–56.

Bowring B.R., 1976, Transformation from spatial to geodetic coordinates, *Survey Review*, 23, 181: 323–327.

Bowring B.R., 1985, The accuracy of geodetic latitude and height equations, *Survey Review,* 28, 218:202–206.

Fukushima T., 1999, Fast transform from geocentric to geodetic coordinates, *Journal of Geodesy,* 73:603–610.

Lapaine M., 1991, A new direct solution of the transformation problem of Cartesian into ellipsoidal coordinates, In *Determination of the geoid,* IAG Symposia, Vol. 106, Springer, Berlin, Heidelberg, New York, 395 –404.

Gerdan, G.P. and Deakin, R.E. 1999, Transforming Cartesian coordinates to geographical coordinates, *The Australian Surveyor,* 44, 1: 55-63.

Guo, G., 2001, A fast procedure for geodetic coordinate transformation, *Bulletin of Surveying and Mapping*, 11:12-13.

Groten, E., 1979, *Geodesy and the earth's gravity field,* Bonn, Dummlers Pubs, 1979-1980, 152-154.

Jones, G.C., 2002, New solutions for the geodetic coordinate transformation, *Bulletin Geodesique*, 76:437-446.

Lin K.C. and Wang J. 1995, Transformation from geocentric to geodetic coordinates using Newton's iteration, *Bulletin Geodesique*, 69:14-17.

Ozone, M.I., 1985, on-Iterative Solution of the ϕ Equation, *Surveying and Mapping*, 45, 2: 169-171.

Pollard, J., 2002, Iterative vector methods for computing geodetic latitude and height from rectangular coordinates, *Bulleting Geodesique*, 76:36-40.

Seemkooei, A.A., 2002, Comparison of different algorithms to transform geocentric to geodetic coordinates, *Survey Review,* 36, 286: 627-633.

Vermeille, H., 2002, Direct transformation from geocentric coordinates to geodetic coordinates, *Bulleting Geodesique*, 76:451–454.

You, R., 2000, Transformation of Cartesian to geodetic coordinates without iterations, *Journal of Surveying Engineering*, 1-7.