

GPS EASY Suite II

easy14—EGNOS-Aided Aviation

KAI BORRE
AALBORG UNIVERSITY

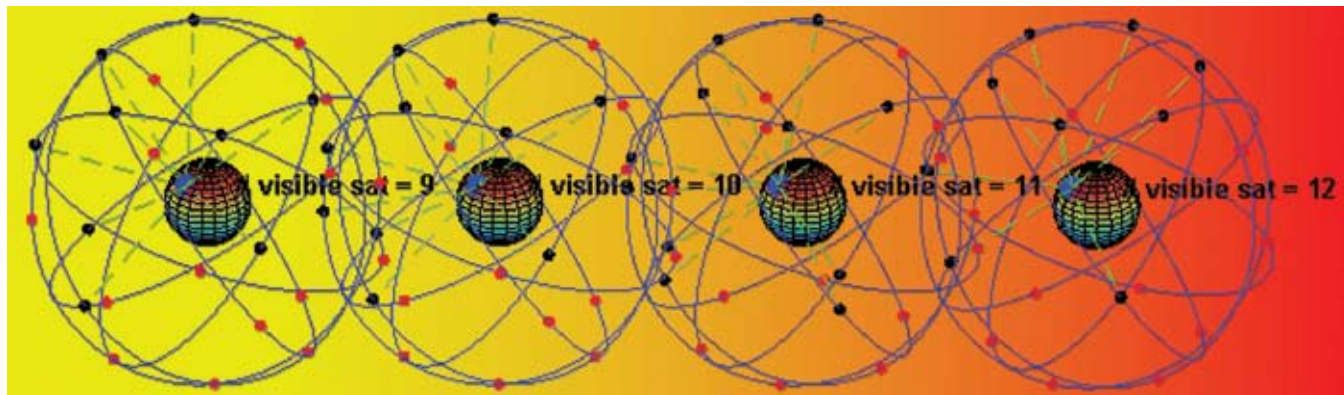


Image of GPS constellation based on public domain file from Wikimedia Commons

In this installment of the series, the author uses Matlab to illustrate aspects of aviation navigation using EGNOS signals to augment GPS.

Some GPS applications generate long time series of position estimates. Most often, an easy way of getting a quick overview on the recorded data is to plot them. However, it may be difficult from a plot to quantify statistical measures such as *mean value*, *standard deviation*, and *circular error probable* (CEP).

easy14 continues the aviation-oriented discussion begun in the July/August 2009 issue of *Inside GNSS*, drawing on use of the European Geostationary Navigation Overlay Service (EGNOS), a satellite-based augmentation system (SBAS). The files for the corresponding MATLAB codes can be found online; details included in the Additional Resources section near the end of this article.

Our demonstration sample uses data collected on August 20, 2008, over a 4.2-hour period with a static GPS L1 receiver with 12 code and carrier channels and two optional SBAS channels, located at a site south of Aalborg, Denmark. This receiver is a commercial one capable of handling EGNOS data. However, dedicated EGNOS receivers exist that exploit the data optimally and from which better-looking plots can be generated.

The EGNOS-corrected positions were computed by using a MATLAB implementation done at the Danish GPS Center (DGC). For a given alert limit a , EGNOS provides an integrity measure that tells the user to use or not to use the position in question.

The EGNOS corrections can be obtained in several ways: directly via the geostationary satellites listed in **Table 1**, or from a service called Signal in Space through the Internet (SISnET),

or through other services. In the present case we apply a post-processing mode which relies on the EGNOS Message Server (EMS) on-line at <ftp://131.176.49.48/pub/>

In polar regions users may have difficulties in receiving signals from the geostationary satellites and, hence, the Internet version becomes useful. However, the Internet is not available near the poles too often!

We shall make a small digression to investigate the magnitude of the elevation angle h of a (geostationary) satellite as a function of the receiver site. Let a ground receiver have coordinates (φ_E, λ_E) and let the sub-satellite coordinates of any satellite be (φ_S, λ_S) . The north pole, the receiver position E , and the sub-satellite point of S make a spherical triangle on a unit sphere with sides $90^\circ - \varphi_E$, $90^\circ - \varphi_S$, and the intersecting angle $\lambda_E - \lambda_S$.

Applying the spherical cosine law on this triangle yields the following expression for the latitude φ reduced to the meridian of E

$$\cos \varphi = \cos \varphi_E \cos \varphi_S \cos(\lambda_E - \lambda_S) + \sin \varphi_E \sin \varphi_S.$$

Specializing to a geostationary satellite with $\varphi_S = \varphi_G = 0$ gives

$$\cos \varphi = \cos \varphi_E \cos(\lambda_E - \lambda_G). \quad (1)$$

Our data are collected at $(\varphi_E, \lambda_E) = (57^\circ, 10^\circ)$; if we received the EGNOS corrections from the AOR-E (PRN 120) satellite, we would have

$$\cos \varphi = \cos 57^\circ \cos(10^\circ - (-15.5^\circ))$$

or a reduced latitude $\varphi = 60.6^\circ$.

Next we assume that our site E and the geostationary satellite G are in the same meridian plane. The Earth radius is $a_E = 6700$ kilometers and the distance between the geostationary satellite and the Earth center is $a_G = 42486$ kilometers. With reference to **Figure 1** we get, in a spherical approximation, for triangle OPG

$$\frac{a_G}{\sin(\pi/2 + h)} = \frac{a_E}{\sin(\pi/2 - (h + \varphi))}. \quad (2)$$

Some rewriting involving the sum relation yields

$$h = \arctan\left(\frac{\left|\cos \varphi - \frac{a_E}{a_G}\right|}{\sin \varphi}\right) \quad (3)$$

The elevation angle h equals zero for $\varphi = 80.9^\circ$; for $\varphi = 60.6^\circ$ we get $h = 20.9^\circ$, (refer to **Figure 2**).

The EGNOS information allows us to correct the observed pseudoranges for atmospheric delays and other error sources. A fast EGNOS correction is computed from navigation messages 2–5 that contain the so-called fast data set. The message starts with the 2-bit issue of data PRN (IODP), an 8-bit preamble, a 6-bit message type identifier (values can be 2, 3, 4, or 5), and a 2 bit issue of data fast (IODF).

Next follows the fast data set for 13 satellites: 12 bits for the fast pseudorange correction PRC and 4 bits for the user *differential range error indicator* (UDREI). The message ends with 24 parity bits, in total 250 bits. The fast data set for the next 13 satellites is contained in the following message and so forth.

The PRC value is given in the interval $[-256.000, +255.875]$ with a resolution of $\frac{1}{8}$ meter.

The range-rate correction RRC of the fast correction is computed as the difference between the current PRC and the previous one divided by the time interval between the two values. The time of applicability t_{of} is identical to the transmission time of the first bit in the message.

The fast correction to the pseudorange for a given satellite identified by its pseudorandom noise code number (PRN) is

$$PR_{corrected}(t) = PR_{observed}(t) + PRC(t_{of}) + RRC(t_{of})(t - t_{of}). \quad (4)$$

The RRC computation must time-out if there is no PRC for eight seconds.

The complete computation of the EGNOS correction involves calculation of an ionospheric correction and a possible tropospheric correction. The variance of the i th observation is composed of four variances: a fast and long term contribution σ_{flt}^2 , a tropospheric correction σ_{tropo}^2 , an ionospheric correction σ_{iono}^2 , and finally receiver noise and multipath σ_{air}^2 :

$$\sigma_i^2 = \sigma_{flt}^2 + \sigma_{tropo}^2 + \sigma_{iono}^2 + \sigma_{air}^2. \quad (5)$$

Typical values for a satellite with high elevation are $\sigma_{flt} = 0.26$ m, $\sigma_{tropo} = 0.01$ meter, $\sigma_{iono} = 0.21$ meter, and $\sigma_{air} = 3.26$ meter, and for a low elevation satellite $\sigma_{flt} = 0.55$ meter, $\sigma_{tropo} = 0.10$ meter, $\sigma_{iono} = 0.91$ meter, and $\sigma_{air} = 3.28$ meter.

Figure 3 shows the time series (deviations from mean values) for the three coordinates East, North, and Up (E, N, U) at

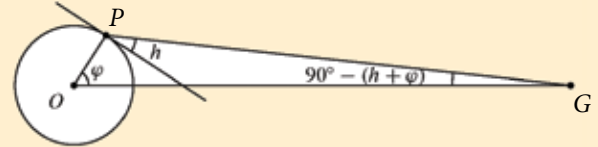


FIGURE 1 Elevation angle h to geostationary satellites as seen from the Earth's surface as a function of latitude φ , spherical approximation

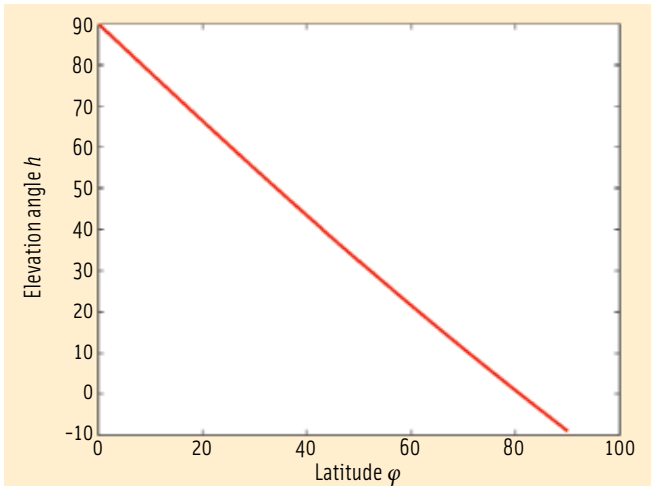


FIGURE 2 Plot of elevation angle h to geostationary satellites as seen from the Earth's surface as a function of latitude φ

Satellite Name	Longitude l_G	PRN
AOR-E	15.5° W	120
Artemis	21.5° E	124
IOR-W	25.0° E	126

TABLE 1. EGNOS geostationary satellites

the location E . Positions computed from the raw pseudoranges (black line) plot as noisy curves.

Next we applied a MATLAB EGNOS correcting code, developed at DGC, to the raw observations; the result is the less noisy curves (red). No smoothing is applied. A further quantification of the graphs in **Figure 3** are added as **Table 2**. **Figure 4** contains a plot of the corresponding horizontal positions.

We have created a script easy141 that gives a recipe for computing *horizontal protection levels* (HPLs) and *vertical protection levels* (VPLs) for aircraft approaches to airports. We start from the coefficient matrix A for a single epoch

$$A = \begin{bmatrix} \frac{X^1 - X}{\rho^1} & \frac{Y^1 - Y}{\rho^1} & \frac{Z^1 - Z}{\rho^1} & 1 \\ \frac{X^2 - X}{\rho^2} & \frac{Y^2 - Y}{\rho^2} & \frac{Z^2 - Z}{\rho^2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{X^s - X}{\rho^s} & \frac{Y^s - Y}{\rho^s} & \frac{Z^s - Z}{\rho^s} & 1 \end{bmatrix} \quad (6)$$

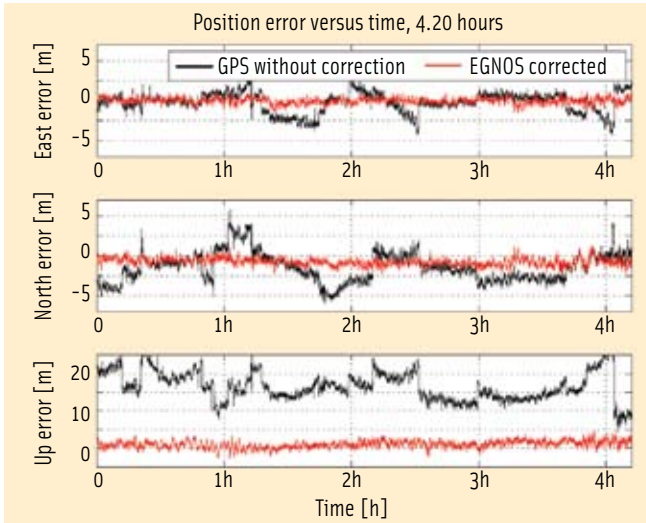


FIGURE 3 4.2 hours of data collected by a static GPS/SBAS receiver. The figure shows east, north, and up as computed from raw pseudoranges (black) and from DGC EGNOS implemented correcting code (red)

	CEP 50% [m]	CEP 95% [m]	s_e [m]	s_n [m]	$mean_e$ [m]	$mean_n$ [m]
Raw Pseudoranges	1.8	3.8	1.3	1.8	-0.2	-1.4
DGC EGNOS-corrected	0.5	1.1	0.4	0.5	0.0	-0.7

TABLE 2. Circular errors probably (CEP), standard errors, and mean values of position for raw pseudoranges collected by a commercial receiver. Also shown are EGNOS corrections as computed by DGC MATLAB code. The numbers are derived from the 4.2-hour data series shown in Figure 3. The cut-off angle is 5° elevation.

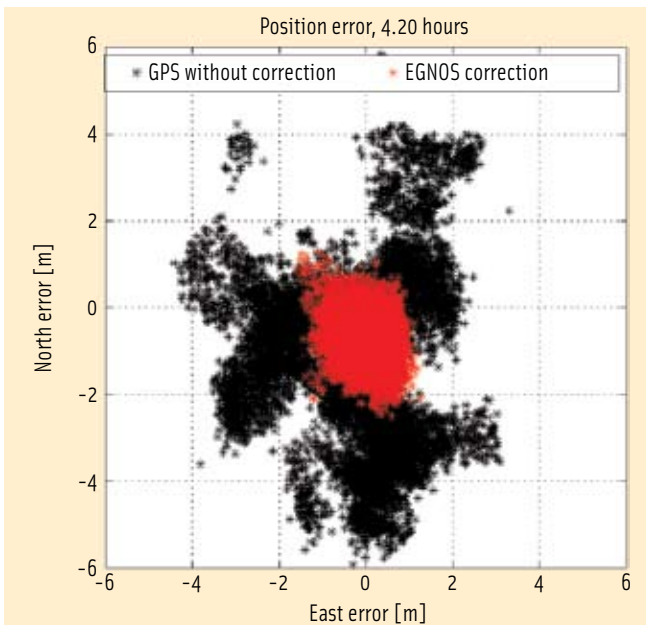


FIGURE 4 4.2 hours of data collected by a Thales DG16 receiver. The figure shows horizontal positions in meters as computed from raw pseudoranges (black) and from DGC EGNOS implemented correcting code (red)

The necessary details may be found in the script.

The larger script easy14 starts from a transformed version of A. The main change is to observe that $\frac{x^1 - x}{\rho^1} = \cos el^1 \cos az^1$ where el^1 means the elevation angle and az^1 the azimuth of PRN i as seen from the receiver position.

The first three columns contain the so-called Euler angles between the line-of-sight and the (X,Y,Z) axes. However, we need the corresponding angles between the line-of-sight and the topocentric (E,N,U) system. The three columns are transformed from (X,Y,Z) to (E,N,U) by post-multiplication with F^T . For further discussion of this point, see equation (9) in easy13.

The product $B = A(:, 1 : 3) F^T$, augmented with a column of ones, can also be computed directly as

$$B = \begin{bmatrix} \cos el^1 \cos az^1 & \cos el^1 \sin az^1 & \sin el^1 & 1 \\ \cos el^2 \cos az^2 & \cos el^2 \sin az^2 & \sin el^2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \cos el^s \cos az^s & \cos el^s \sin az^s & \sin el^s & 1 \end{bmatrix} \quad (7)$$

The next steps are described by simple expressions but involve heavy computational burdens. A weight matrix W for the observed pseudoranges is defined in terms of the variances σ_i^2 :

$$W = \text{diag} [\sigma_1^{-2} \quad \sigma_2^{-2} \quad \dots \quad \sigma_s^{-2}], \quad (8)$$

$$\Sigma = (B^T W B)^{-1}.$$

The 2×2 block matrix of the covariance matrix Σ containing the entries $\Sigma_{11} = \sigma_1^2$, $\Sigma_{22} = \sigma_2^2$, and $\Sigma_{12} = \Sigma_{21} = \sigma_{12}$ constitute the coefficients in a quadratic form

$$\sigma_1^2 x^2 + 2\sigma_{12} xy + \sigma_2^2 y^2 = 1.$$

This quadratic form defines a confidence ellipse. The largest eigenvalue λ_1 of this quadratic form is the semi-major axis of the confidence ellipse for the position solution

$$\lambda_1 = \frac{1}{2} (\sigma_1^2 + \sigma_2^2 + \sqrt{(\sigma_1^2 + \sigma_2^2)^2 - 4(\sigma_1^2 \sigma_2^2 - \sigma_{12}^2)}),$$

Equation (9.78) in the book by G. Strang and K. Borre, listed in Additional Resources, provides further context for the preceding step.

The expression for λ_1 can be rewritten as

$$\lambda_1 = \frac{\sigma_1^2 + \sigma_2^2}{2} + \sqrt{\left(\frac{\sigma_1^2 - \sigma_2^2}{2}\right)^2 + \sigma_{12}^2}. \quad (9)$$

This version is found in the minimum operational performance standards (MOPS) for GPS/Wide Area Augmentation System (WAAS) airborne equipment published in RTCA/DO-229C, page J.1.

The EGNOS standard defines two magnification factors $K_H = 6$ and $K_V = 5.33$. Remember that $\Sigma_{33} = \sigma_{33}$ and finally we obtain:

$$\text{HPL} = K_H \sqrt{\sigma_1}$$

$$\text{VPL} = K_V \sigma_{33}$$

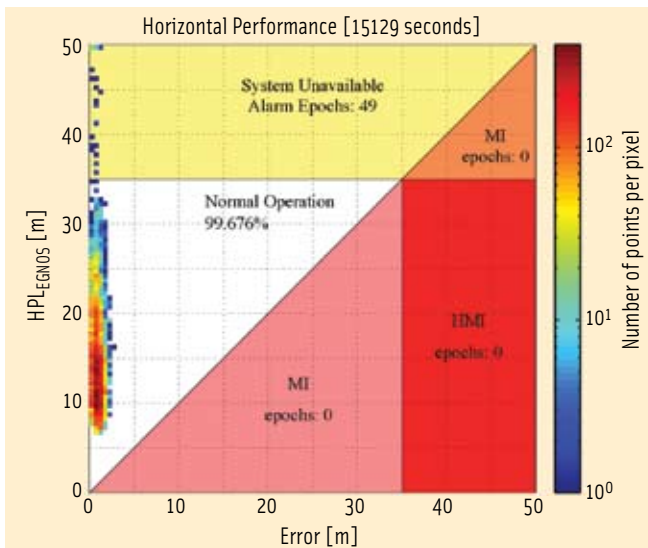


FIGURE 5 Positions depicted in Figure 3 are plotted versus horizontal protection levels. Integrity fails in the region marked HMI, hazardous misleading information. In the upper area marked MI, integrity fails and the system is unavailable. HPL failures happen in the lower area marked MI. In short: *USE* positions depicted in the white area, *DO NOT USE* positions depicted in colored areas

Typical values are: HPL = 15 and VPL = 6. The EGNOS document also introduces its own non-precise HPL value, which is 1.03 times larger than the HPL used here. Everything is implemented in easy141.

Contemporary positioning theory uses the following four concepts that enter into almost every aviation specification *accuracy, integrity, continuity, and availability*.

The first concept, accuracy, fits our intuition well as it measures the difference between the (corrected) computed position and the true position. An SBAS implementation is obliged to quantify the accuracy of a wide-area differentially corrected position solution. Accuracy may be estimated by the difference between a receiver position and the true position which, for CEP 95%, is only exceeded 5 percent of the time in the absence of system failures.

Integrity risk is defined as the probability that the SBAS exceeds either the horizontal or vertical alert limits (HALs or VALs) *and* the system alert is silent beyond the time-to-alarm. EGNOS has been designed for a six-second time-to-alarm.

Continuity and availability are expressed in global terms. If a user experiences a system outage due to signal blockage, from a global perspective this is not a loss in continuity of the system — it is a local phenomenon.

Clearly, we need to distinguish between issues related to availability and continuity and what is availability of the system from a user's perspective.

A sophisticated plot was developed at Stanford University in 1998. The histogram of **Figure 5** reports the horizontal system metrics provided by an EGNOS implementation done at DGC for a static user at Aalborg.

The EGNOS concept only involves an alert limit *a*. Knowing the value of *a* we can draw a half line from origin through

(*a, a*) which partly limits the white area in which the attractive bins must lie!

The horizontal axis indicates the error $\sqrt{e^2 + n^2}$ in the SBAS navigation solution with respect to the surveyed antenna location. The vertical axis indicates the protection level computed for each and every position solution.

Each bin tabulates the number of occurrences of a specific pair (*x, y*) = (error, protection level), and the color indicates the total number of epochs in which that pair occurs. Note that the color scale is logarithmic and the bins are quantized into 0.25-meter squares.

In case the HPL is larger than the alarm limit *a*, we experience an integrity failure. The true error should always be less than the HPL. The long-term availability requirement of the SBAS is 99.9 percent and, hence, at least 999 out of 1,000 points should lie within the “Normal Operation” (USE) region.

In the Figure 5 results, the system maintained 99.7 percent availability in horizontal positioning and the system was unavailable in 49 epochs out of 15,129 that is 0.3 percent; note that (99.7 + 0.3) % = 100%.

Availability can only be associated with geostationary satellites. For users employing the Internet to obtain the EGNOS information, everything changes. In the present study we use EMS and not the real-time service SISNeT.

Figure 6 (on page 72) shows the vertical system performance corresponding to the horizontal data presented previously. The SBAS correction in the vertical dimension has poorer performance than the horizontal due to its weaker geometry; the alert limits are *a* = 12, 20 meters. The 12-meter level corresponds to Category I landing and the 20-meter level to instrument precision with vertical guidance (IPV). In this scenario, the positioning system met all three safety metrics (accuracy, integrity, and continuity) with an availability of 99.061 percent.

Acknowledgment

Kostas Dragūnas added several changes to the original M-files vplstat and hplstat written by Todd Walter of Stanford University. The files are included in the Easy Suite II files (see Additional Resources) with the permission of Stanford University. Users also need the bound2 file in the download directory.

Manufacturers

The data in easy14 was generated using a Thales DG 16 GPS receiver, from **Magellan Navigation S.A.S.**, Santa Clara, California, USA.

Additional Resources

The original Easy Suite can be found online at <<http://kom.aau.dk/~borre/easy/>>. The complete set of Easy Suite II Matlab codes can be found in compressed (“zipped”) files at <<http://gps.aau.dk/~borre/easy2/>>.

RTCA, Inc. (formerly Radio Technical Commission for Aeronautics), *RTCA/DO-229C, Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment*, 2001

GPS EASY Suite II continued on page 72

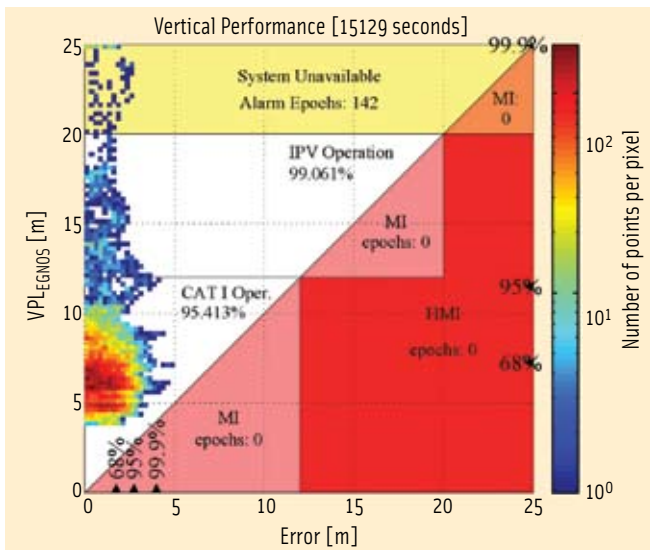


FIGURE 6 Positions depicted in Figure 10 are plotted versus vertical protection levels. Integrity failures happen in the areas marked HMI. The 20 m level corresponds to Instrument Precision with Vertical guidance (IPV). Cat I Oper. corresponds to horizontal and vertical accuracies for precision landing for Category I (decision height 200 feet). The lower and middle regions marked MI designate VPL failures. In the upper region marked MI integrity and unavailability overlap

GPS EASY Suite II continued from page 65

Strang, G., and K. Borre, *Linear Algebra, Geodesy, and GPS*, Wellesley-Cambridge Press, Wellesley, Massachusetts USA, 1997

Corrections to easy13

In the EASY Suite II installment in the July/August 2009 issue of *Inside GNSS*, in Figures 1 and 4, the expression “protection level α ” should have been “protection level α ”

In easy13.m the Matlab code is missing the F matrix (the transformation matrix from ECEF to ENU) in the expression for M. Hence the alpha values and HPL are wrong. Consequently Figure 3 is valid for δx_{XY} and not δx_{EN} as stated on the vertical axis.

On page 49 of the article in the paragraph following equation (10), $M_0 = M(1:2,1:2)$ should be $M_0 = M(1:2,:)$.

Author



Originally graduated as a chartered surveyor, **Kai Borre** obtained his Ph.D. in geodesy from Copenhagen University, and a Doctor of Technology degree from Graz University of Technology. He has been a full professor in geodesy at Aalborg University since 1976.

For more than 30 years Borre has conducted research and performed education in the area of satellite based positioning. In 1996 he established the Danish GPS Center and since 2000 has been head of a two-year international M.Sc. program in GPS technology.

Borre is a coauthor of the widely used textbooks, *Linear Algebra, Geodesy, and GPS* and *A Software-Defined GPS and Galileo Receiver; Single-Frequency Approach*.

Since the early 1990s Borre has published Matlab code for processing of GPS observations, and since 2003 he published Matlab code together with explanatory text, a very successful new pedagogical concept.

