

GPS EASY Suite II

A Matlab Companion

KAI BORRE
AALBORG UNIVERSITY

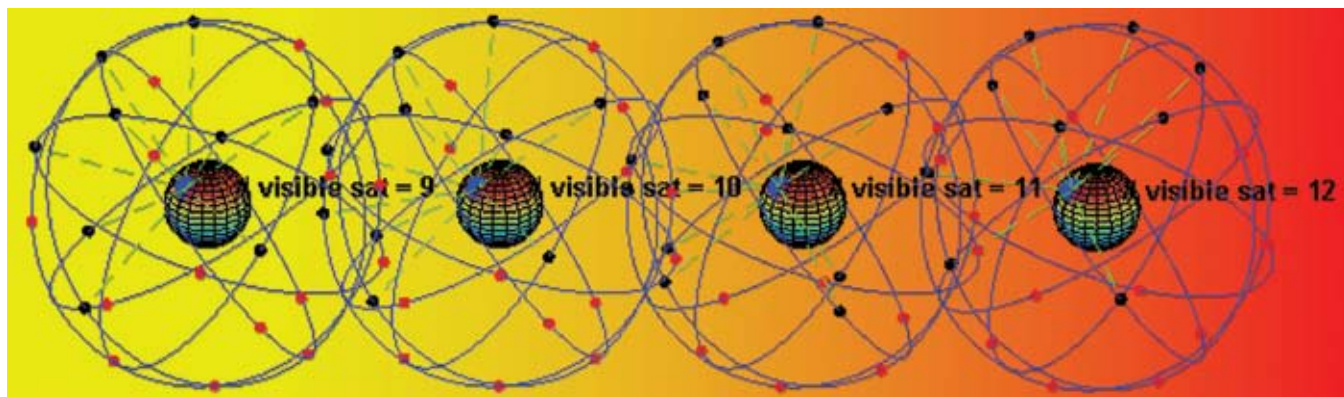


Image of GPS constellation based on public domain file from Wikimedia Commons

Many practitioners in the GNSS field are familiar with Matlab, a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numeric computation. In this new series, a prominent Danish GNSS researcher uses Matlab to illustrate and explain a variety of common GPS issues.

In 2003 I published a paper called “The GPS EASY Suite — Matlab Code for the GPS Newcomer.” The paper consisted of 10 parts, which are cited in the accompanying sidebar, “EASY Suite I Topics.” Each installment included a printed text and a file of related Matlab code or scripts (an *M*-file) that could be downloaded from a designated web site.

Each part contained an answer to often-asked questions in my classes. I did not, however, only offer an answer to the questions, but also a Matlab code that solves the problem. The article became a tremendous success and remained the most downloaded paper from the *GPS Solutions* site for more than a year.

Now, students love this sort of support. But I also received positive reactions from professionals who used the code in research papers, which saved them a lot of coding efforts.

The original Matlab code also turned out to be the most downloaded file from the Aalborg website. It resulted in numerous e-mails from interested readers asking for more files. These requests now answered by the creation of eight additional *M*-files. Some involve more complex problems and coding.

Since 2003, I have received a steady flow of additional questions from readers of Gilbert Strang’s and my book, *Linear Algebra, Geodesy, and GPS*. Last summer I decided to create another set of common GPS problems/solutions and accompa-

nying Matlab code — EASY Suite II. These will be presented in serial fashion, as one or more exercises in this and forthcoming issues of *Inside GNSS*.

Introducing the New Suite

Easy Suite II augments the set of basic computational tasks with the following topics:

EASY11, stereographic sky plot of satellite orbits and plot of time when satellites are above a given local horizon

EASY12, details of the LAMBDA method, explained through a small numerical example

EASY13, receiver autonomous integrity monitoring (RAIM), horizontal protection level (HPL), and vertical protection level (VPL)

EASY14, sample of space-based augmentation system (SBAS) corrected positions and their presentation in Stanford plots

EASY15, accuracy comparison between pseudorange based stand-alone positions, baselines computed using pseudoranges alone, and pseudoranges and carrier phase observations

EASY16, error analysis of a selected one-way observation

EASY17, satellite orbits in inertial and Earth-centered, Earth-fixed (ECEF) systems, and curve defined by sub-satellite points

EASY18, computation of differential corrections at a base station.

The original suite was thought of as comprising the base for an elementary course in GPS while, as reflected in the subjects, the new suite is of a more optional, topical character.

In 2003 and later, many users encountered difficulty in finding the necessary files for running individual EASY-files. Therefore, I chose to copy all files needed into single directories accessible online, so that each directory becomes self-contained. The price we pay is multiple copies of basic *M*-files.

The complete set of Easy Suite II Matlab codes can be found in compressed (“zipped”) files at <http://gps.aau.dk/~borre/easy2>. Readers can find much of the theoretical background for the EASY scripts in Chapters 14 and 15 of *Linear Algebra, Geodesy, and GPS* (see Additional Resources for details). However, we have added some text that emphasizes certain issues that are central in the codes, but may be difficult to find in a textbook.

EASY11: GPS Sky Plots

The EASY-Suite starts with a polar plot of satellite orbits (see **Figure 1**) as viewed from a given location, a graph showing the number of visible satellites, and the period of time when they are visible during 24 consecutive hours.

EASY11 itself is based on an almanac downloaded most easily from the National Geodetic Survey (NGS) website <http://www.ngs.noaa.gov/CORS/Data.html>. The actual file name is brdc1550.08n.

The RINEX file has been reformatted into Matlab’s binary format for satellite ephemerides using the *M*-file *rinexe*. The user enters (ϕ, λ) for the position on the ground where the plot is to be used, and a value for the elevation mask to be set. Then the azimuth and elevation angles for all visible positions of the included satellites are computed and plotted as polar coordinates.

Next follows bookkeeping on how many and which satellites are visible during the day and the time periods when they can be seen. **Figures 2** and **3** show the resulting plots.

The Matlab code is simple, the result is impressive, and useful. In early GPS days when the constellation was incomplete, such plots were especially valuable for planning purposes to be sure that enough satellites would be available for positioning. Except in the most severe terrain and urban canyons, receivers can find plenty of GPS (and GLONASS) satellites around the clock, and this situation will become even more pronounced when Galileo satellites are launched.

EASY12: LAMBDA Method

Most students find the theory behind the Least-squares AMBIGUITY Decorrelation Adjustment (LAMBDA) for ambiguity resolution difficult to understand. We shall try to smooth the path.

First we describe in a subsection the linear algebra involved. Next we add an *M*-script that elucidates how the method works on a simple case with three ambiguities.

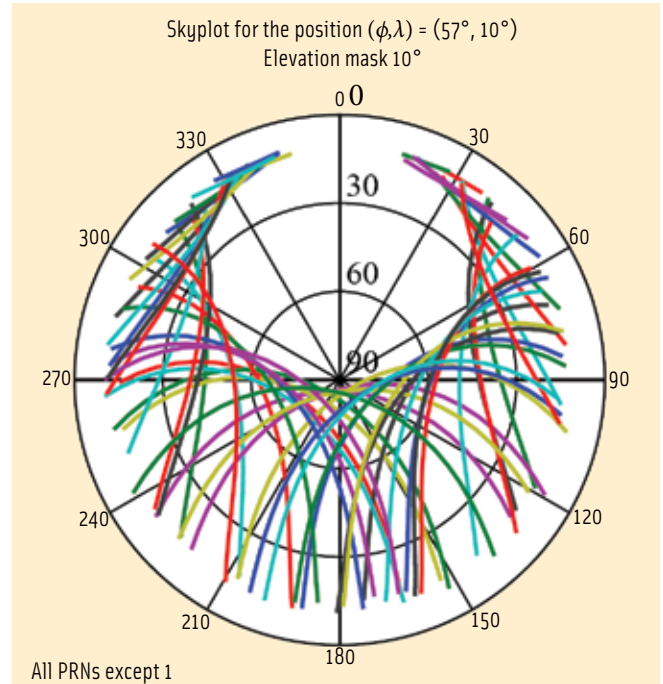


FIGURE 1 Sky plot including all GPS satellites for a period of 24 hours at a given position. The elevation mask is 10 degrees.

EASY Suite I Topics

The original Easy Suite, which can be found online at <http://kom.aau.dk/~borre/easy/>, dealt with the following issues:

- EASY1, time conversion
- EASY2, computation of a satellite’s position from an ephemeris
- EASY3 computation of a receiver’s position from pseudoranges
- EASY4 computation of a baseline from pseudoranges alone
- EASY5 computation of a baseline from pseudoranges and phase observations using a least-squares solution
- EASY6 the same, but now using a Kalman filter for the baseline estimation
- EASY6e the same, but introducing down-weighting of older observations
- EASY7 estimation of receiver clock offset
- EASY8 check of cycle slips and receiver clock reset
- EASY9 various coordinate representations of a given baseline
- EASY10 estimation of ionospheric delay for the individual satellites

Linear Algebra for LAMBDA

Let the vector \mathbf{b} contain the three components of the baseline and the vector \mathbf{a} contain ambiguities for the L_1 frequency and possibly for the L_2 frequency. The double differenced observations are collected in the vector \mathbf{y} :

$$\begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} \mathbf{b} \\ \mathbf{a} \end{bmatrix} = \mathbf{y} + \text{errors}. \quad (1)$$

The normal equations are

$$\begin{bmatrix} B^T B & B^T A \\ A^T B & A^T A \end{bmatrix} \begin{bmatrix} \hat{\mathbf{b}} \\ \hat{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} B^T \\ A^T \end{bmatrix} \mathbf{y} \quad (2)$$

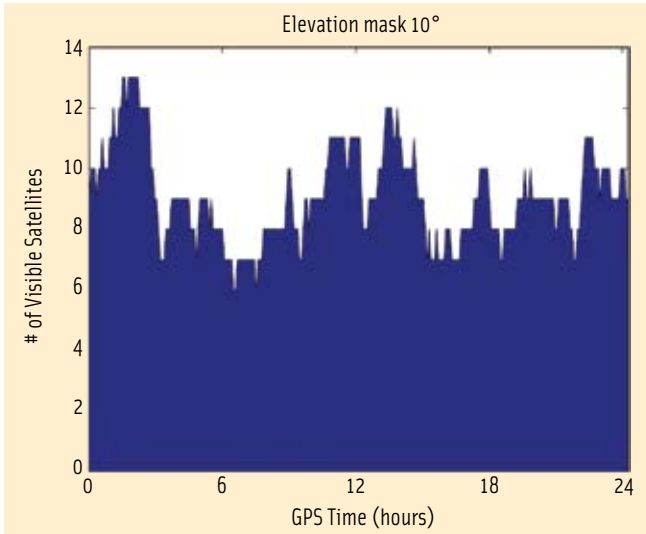


FIGURE 2 Number of visible satellites

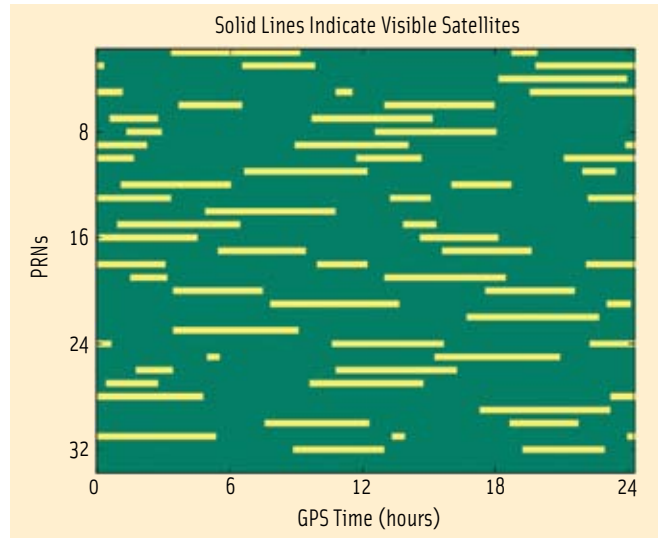


FIGURE 3 Time period when the visible satellites are 10 degrees or higher above the horizon

and the formal solution is

$$\begin{bmatrix} \hat{\mathbf{b}} \\ \hat{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} B^T B & B^T A \\ A^T B & A^T A \end{bmatrix}^{-1} \begin{bmatrix} B^T \\ A^T \end{bmatrix} \mathbf{y} = \begin{bmatrix} Q_{\hat{\mathbf{b}}} & Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} \\ Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}}^T & Q_{\hat{\mathbf{a}}} \end{bmatrix} \begin{bmatrix} B^T \\ A^T \end{bmatrix} \mathbf{y}. \quad (3)$$

The components of the solution vector $\hat{\mathbf{a}}$ are reals; however, we want a solution $\check{\mathbf{a}}$ of integers! Hence, we find an integer vector \mathbf{a} such that

$$(\hat{\mathbf{a}} - \mathbf{a})^T Q_{\hat{\mathbf{a}}}^{-1} (\hat{\mathbf{a}} - \mathbf{a}) = \min. \text{ over integer vectors } \mathbf{a}. \quad (4)$$

After the integer solution $\check{\mathbf{a}}$ is found we substitute it for $\hat{\mathbf{a}}$. Consequently, the solution for $\hat{\mathbf{b}}$ changes to $\check{\mathbf{b}}$. In order to determine $\check{\mathbf{b}}$, we multiply the lower block row in (3) by $Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1}$ and subtract from the upper block row:

$$\begin{bmatrix} \hat{\mathbf{b}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} \hat{\mathbf{a}} \\ \hat{\mathbf{a}} \end{bmatrix} = \begin{bmatrix} Q_{\hat{\mathbf{b}}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} & 0 \\ Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}}^T & Q_{\hat{\mathbf{a}}} \end{bmatrix} \begin{bmatrix} B^T \\ A^T \end{bmatrix} \mathbf{y}. \quad (5)$$

The upper block row gives:

$$\hat{\mathbf{b}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} \hat{\mathbf{a}} = (Q_{\hat{\mathbf{b}}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1}) B^T \mathbf{y}.$$

The right side is known and constant. If we change $\hat{\mathbf{a}}$ to $\check{\mathbf{a}}$, then $\hat{\mathbf{b}}$ changes to $\check{\mathbf{b}}$ and we have:

$$\hat{\mathbf{b}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} \hat{\mathbf{a}} = \check{\mathbf{b}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} \check{\mathbf{a}}$$

or

$$\check{\mathbf{b}} = \hat{\mathbf{b}} - Q_{\hat{\mathbf{b}}\hat{\mathbf{a}}} Q_{\hat{\mathbf{a}}}^{-1} (\hat{\mathbf{a}} - \check{\mathbf{a}}). \quad (6)$$

The right side is known and $\check{\mathbf{b}}$ is quickly found.

Minimizing a Quadratic Expression over Integer

The LAMBDA method starts as other least-squares computations by solving the normal equations (3). The result is partly the vector of real-valued ambiguities $\hat{\mathbf{a}}$ and partly the pertinent covariance matrix $Q_{\hat{\mathbf{a}}}$.

The core problem can be stated in a very clear form

$$\text{minimize } (\mathbf{I} - \hat{\mathbf{I}})^T Q_{\hat{\mathbf{I}}}^{-1} (\mathbf{I} - \hat{\mathbf{I}}) \text{ over integer vector } \mathbf{I}.$$

In order to emphasize the integer nature of the problem we have swapped the notation from \mathbf{a} to \mathbf{I} .

This is the problem we study, and in one case this is especially simple. If $Q_{\hat{\mathbf{I}}}^{-1}$ is diagonal, the best vector comes from rounding each component of $\hat{\mathbf{I}}$ to the nearest integer.

The components are uncoupled when $Q_{\hat{\mathbf{I}}}^{-1}$ is diagonal. The quadratic is purely a sum of squares $\sum (Q_{\hat{\mathbf{I}}}^{-1})_{jj} (\mathbf{I}_j - \hat{\mathbf{I}}_j)^2$. The minimum comes by making each term as small as possible. So the best $\hat{\mathbf{I}}_j$ is the integer nearest to $\hat{\mathbf{I}}_j$.

In practice the individual $\hat{\mathbf{I}}_j$ components are highly correlated. Recalling that we often deal with 20 ambiguities, we realize that a search procedure to find the minimum is unrealistic. Therefore, an idea about decorrelating the ambiguities as much as possible, before starting the search, should lead to an effective procedure.

This is just what the LAMBDA method does. It was described by Peter Teunissen in 1993 and is, from a theoretical point of view, still considered to be the best method.

The first step consists in transforming $Q_{\hat{\mathbf{I}}}^{-1}$ such that its off-diagonal entries become numerically smaller. These entries measure the correlation.

We start by computing the LDL^T decomposition of the given covariance matrix

$$Q_{\hat{\mathbf{I}}} = LDL^T.$$

Next we construct a matrix of integers Z from L by a sequence of integer Gauss transformations and permutations such that

$$Q_{\hat{\mathbf{I}}} = Z^T Q_{\hat{\mathbf{I}}} Z$$

is as “diagonal” as possible. Now the search defined by (4) is substituted by a search over integers \mathbf{I} :

$$(\hat{\mathbf{I}} - \mathbf{I})^T \mathbf{Z}^T \mathbf{Q}_I^{-1} \mathbf{Z} (\hat{\mathbf{I}} - \mathbf{I}) = \min. \quad (7)$$

The choice of \mathbf{Z} is based on integer elimination starting with the first row of \mathbf{L} . This will certainly involve row exchanges and, therefore, \mathbf{Z} will not be triangular. The essential idea for this *lattice basis reduction* was given by A. K. Lenstra et alia in 1982 (See Additional Resources), and the algorithm is sometimes called L^3 . For further details see Strang and Borre, pages 495–499.

The result of the search is the integer vector $\hat{\mathbf{I}}$. Finally, $\hat{\mathbf{I}}$ is transformed back to the ambiguity space according to

$$\bar{\mathbf{I}} = (\mathbf{Z}^{-1})^T \hat{\mathbf{I}}. \quad (8)$$

Note that both \mathbf{Z} and \mathbf{Z}^{-1} must have integer entries. A consequence of this is that $\det(\mathbf{Z}) = 1$ as seen from Cramer's rule. The condition on the determinant of \mathbf{Z} means that the \mathbf{Z} transformation preserves the search volume. At the end of this process, we have transformed a highly correlated space (elongated ellipses) into a sphere-like search space, which diminishes the search time tremendously.

EASY12 illustrates the computational steps including a few numerical details. We choose a numerical example from P. J. de Jonge and C. C. J. M. Tiberius (1996). Further computational details are well described in that report.

A Numerical Example

Given the float ambiguities

$$\hat{\mathbf{I}} = \begin{bmatrix} 5.45 \\ 3.1 \\ 2.97 \end{bmatrix} \quad (9)$$

with covariance matrix

$$\mathbf{Q}_I = \begin{bmatrix} 6.290 & 5.978 & 0.544 \\ 5.978 & 6.292 & 2.340 \\ 0.544 & 2.340 & 6.288 \end{bmatrix}. \quad (10)$$

We introduce integer shifts of $\hat{\mathbf{I}}$ in order to secure that $-1 < \hat{\mathbf{I}}_i \leq 1$:

$$\begin{bmatrix} 5 \\ 3 \\ 2 \end{bmatrix}$$

This leaves the remainders

$$\hat{\mathbf{I}}_r = \begin{bmatrix} 0.45 \\ 0.10 \\ 0.97 \end{bmatrix}.$$

Now we factorize \mathbf{Q}_I into LDL^T with the lower triangular matrix

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 1.065 & 1 & 0 \\ 0.087 & 0.372 & 1 \end{bmatrix} \quad (11)$$

and the diagonal matrix

$$\mathbf{D} = \begin{bmatrix} 0.090 & 0 & 0 \\ 0 & 5.421 & 0 \\ 0 & 0 & 6.288 \end{bmatrix}. \quad (12)$$

We then compute the initial size of the search ellipsoid as the squared distances of partially rounded float vectors to the float vector in the metric of the covariance matrix. In the present example the search volume is determined by $\chi^2 = 1.245$.

The call $[Q_r, Z, L_t, D_t] = \text{decorrel}(Q_I, \hat{\mathbf{I}}_r)$ computes the integer transformation matrix \mathbf{Z} , a decorrelated covariance matrix \mathbf{Q}_r , and the transformed version of the LDL^T decomposition. All quantities are mentioned below:

$$\mathbf{Z} = \begin{bmatrix} -2 & 3 & 1 \\ 3 & -3 & -1 \\ -1 & 1 & 0 \end{bmatrix}. \quad (13)$$

The transformed and shifted ambiguities are

$$\mathbf{I}_s = \mathbf{Z}^T \hat{\mathbf{I}}_r = \begin{bmatrix} -1.57 \\ 2.02 \\ 0.35 \end{bmatrix}. \quad (14)$$

The transformed, decorrelated covariance matrix is

$$\mathbf{Q}_r = \mathbf{Z}^T \mathbf{Q}_I \mathbf{Z} = \begin{bmatrix} 4.476 & 0.334 & 0.230 \\ 0.334 & 1.146 & 0.082 \\ 0.230 & 0.082 & 0.626 \end{bmatrix}. \quad (15)$$

Note the diminished off-diagonal terms! The lower triangular \mathbf{L}_t used in the search is

$$\mathbf{L}_t = \begin{bmatrix} 1.000 & 0.000 & 0.000 \\ 0.268 & 1.000 & 0.000 \\ 0.367 & 0.131 & 1.000 \end{bmatrix},$$

and the diagonal matrix \mathbf{D}_t used in the search is

$$\mathbf{D}_t = \begin{bmatrix} 4.310 & & \\ & 1.135 & \\ & & 0.626 \end{bmatrix}.$$

Determining the size of the search volume for the transformed \mathbf{L}_t and \mathbf{D}_t yields $\chi^2 = 1.218$. The search domain is defined as $(\mathbf{I} - \hat{\mathbf{I}})^T (\mathbf{Z}^T \mathbf{Q}_I \mathbf{Z}) (\mathbf{I} - \hat{\mathbf{I}}) < \chi^2$, for \mathbf{I} integer. The final and fixed ambiguities are

$$\bar{\mathbf{I}} = \begin{bmatrix} 5 \\ 3 \\ 4 \end{bmatrix}.$$

Comparing this vector with $\hat{\mathbf{I}}$ in (9), we notice that the final integer result could not be derived from the float values; this is due to the non-zero correlations between the individual ambiguities.

With this we end the simple numerical example demonstrating the mechanism of the LAMBDA method.

In the next issue of this periodical, we will illustrate the concept of receiver autonomous integrity monitoring (RAIM), and

horizontal and vertical protection levels as applied especially in aviation.

Manufacturers

Matlab is a product of **The MathWorks, Inc.**, Natick, Massachusetts, USA.

Additional Resources

[1] de Jonge, P. J., and C. C. J. M. Tiberius, "The LAMBDA Method for Integer Ambiguity Estimation: Implementation Aspects," *Delft Geodetic Computing Centre LGR series, No. 12*, Delft, Netherlands, 1996

[2] Lenstra, A. K., and H. W. Lenstra, Jr., and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen* 261 (4): 515–534, (1982)

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

[4] Teunissen, P. J. G., "The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation," *Journal of Geodesy*, 70: 65–82, 1995


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. 

GNSS World continued from page 17

GPS produced a one-meter URE in 2008, Buckman said. The GPS III satellites, which will carry the new civil LIC signal, are designed to have a URE that is four times better.

Galileo. Turning at last to Europe's Galileo, the laborious process of contracting out the fully operational capability (FOC) system development continues. In Munich, Fotis Karamitsos, European Commission director-general for transport and energy, and Paul Verhoef, head of the Galileo unit, indicated that agreements with companies winning the lead contracts for six work packages should be signed between September and the end of this year.

Discussions at the Summit revealed tensions around negotiations with China about a frequency overlay of Compass signals on the security-oriented Public Regulated Service as well as the question of whether the costs to build Galileo can be kept within the €3.4-billion limit agreed by the European Council and the European Parliament.

In answer to a question at the March 3 opening plenary, Karamitsos insisted that "we have no reason to believe that FOC won't be delivered on time and on budget."

Responding to a comment that "several member states" and private companies have already suggested creating a "light" version of Galileo — fewer services, signals, and/or satellites, Karamitsos said he that the European Union (EU) member states have a "legal obligation to deliver the full system. Galileo satellites will be acquired in blocks of 10, 8, and 8.

Karamitsos complained of "people negotiating through the press," adding, "In this time of economic constraints it doesn't make sense for our industry to try to make money over" the amount allocated for the program.

According to one European source, the reference was to Surrey Satellite Technology Ltd. (SSTL), a UK firm whose acquisition by EADS Astrium closed in January as well as EU mem-

bers uninterested in using the PRS. SSTL, which specializes in smaller, economical satellite designs, built Galileo's GIOVE-A satellite now in orbit.

SSTL, along with its bidding partner OHB System AG (OHB), has been short-listed as a candidate for the Galileo FOC space segment (with EADS as the other contender) and are preparing for the submission of a "refined proposal" to the European Space Agency.

Versus Compass. Meanwhile, the issue of the Compass/Galileo signal overlay — which recalls an earlier attempt to overlay the PRS on the GPS M-code — continues unresolved after two meetings between Chinese and EC representatives. Some years ago, China attempted unsuccessfully to gain access to the encrypted PRS, which requires unanimous agreement of EU member states before a non-EU nation can do that.

"PRS needs spectral separation," insisted Paul Verhoef, head of the EC's unit for Galileo and intelligent transport, who acknowledged that negotiations with China are "going slower than we hoped."

China's ambitious launch schedule, which requires final decisions on Compass's frequency plan, increases the urgency of the dialog. "We hope to get agreement [with Galileo] before we launch, but we cannot wait to do the validation and development of the system," Jing said in response to a question from the Munich audience.

The situation reflects the ill will that has arisen since the two sides signed agreements in 2003 and 2004 to cooperate on Galileo, including a €200-million Chinese contribution to program development.

In the session on competition among GNSS systems, Yin said that China's industry had found it hard to compete for contracts in the Galileo FOC procurement, even though the nation had allocated €70 million for the in-orbit validation (IOV) phase. "Several IOV cooperation projects could not be implemented smoothly, due to obstacles and barriers," he added. 