

Milan Horemuž
Johan Vium Andersson

Polynomial interpolation of GPS satellite coordinates

Published online: 14 January 2006
© Springer-Verlag 2006

M. Horemuž (✉)
Division of Geodesy, Royal Institute
of Technology, Drottning Kristinas vag 30,
Stockholm, Sweden
E-mail: horemuz@kth.se
Tel.: +46-8-7907335
Fax: +46-8-7907343

M. Horemuž · J. V. Andersson
Department of Infrastructure,
Royal Institute of Technology, Drottning
Kristinas vag 30, Stockholm, Sweden

Abstract This article describes an algorithm for polynomial interpolation of GPS satellite coordinates and its implementation in MATLAB. The algorithm is intended for real-time processing software and computes the position and velocity of GPS satellites from both broadcast and precise ephemerides. Tests with different orders of polynomials, and with different time spans used for polynomial fitting, show suitable settings with respect to the required interpolation precision.

Introduction

The computation of satellite positions is a fundamental task in all GPS positioning software. The data needed for this computation can come in the form of a broadcast or a precise ephemeris. The broadcast ephemeris is available from the GPS, as a set of parameters is sent to the user via the navigation message. The parameters are updated by the control center quite frequently, approximately every 2 h, and the accuracy of the computed coordinates is about 3 m. There are several types of precise ephemerides produced by various agencies in the world based on data from permanent GPS sites. Their accuracy ranges from about 0.2 m for predicted to about 0.05 m for a final, post-processed orbit. Precise ephemerides are often distributed in SP3 format, where the coordinates and satellite clock errors for all GPS satellites are usually listed at 15-min intervals. In this article, we refer to these types of ephemerides as *tabular orbits*. Tabular orbits can be created from the broadcast ephemeris simply by computing the coordinates of all

the satellites at some time interval. The user then needs to employ some interpolation algorithm to compute the coordinates for any other epoch covered by the given tabular orbits. The most rigorous approach for the interpolation is solving the differential equation of satellite motion in the perturbed gravitational field, taking into account solar radiation pressure. This approach is described, for example, in Hugentobler et al. (2001) and is used mostly for orbit determination. Given “solved” orbits in the form of an SP3 precise ephemeris, this approach is unnecessarily complicated. Simple polynomial or trigonometric interpolation (using an SP3 file with a reasonably short time interval) yields interpolated coordinates with centimeter or better accuracy. These two types of orbit interpolation are reviewed in Schenewerk (2003). He shows that both types are equally accurate for interpolation, but trigonometric functions give better results for extrapolation. We should point out that satellite coordinate computation by fitting a polynomial or trigonometric function gives reasonable accuracy only within the fit region.

Software design

Our implementation is intended for real-time applications, which means that the orbit parameters are not always available for the whole measurement period in advance. Therefore, it is necessary to create a process, or module, that checks for ephemeris availability and performs updates. In our implementation, this orbit module creates a so-called *standard ephemeris*, which is a set of polynomial coefficients for coordinate and clock interpolation. There is one set of coefficients per specified time span. The processing module then calls a function that reads the coefficients valid for given epoch and computes satellite coordinates. In this way, the processing module can be written generally for both types of ephemerides. Figure 1 shows the flow diagram of the orbit module.

For testing and debugging purposes, or for post-processing applications, it is not necessary to check continuously for a new ephemeris. The ephemeris is read in either at the beginning from a file in RINEX format in the case of the broadcast ephemeris, or in SP3 format in the case of the precise ephemeris. If we use a broadcast ephemeris, we have to first convert it to the tabular form and then compute a standard ephemeris using the same routine as for the precise ephemeris. The creation of the tabular form involves the computation of each satellite's coordinates and clock corrections at a certain time interval, which by default is 15 min. For this computation, we apply the standard algorithm described in the ICD-GPS-200C document (ARINC 2000).

Standard orbits

Here, the term “standard orbits” means sets of polynomial coefficients; one set per chosen time period. The user can choose the order of polynomial, as well as the fit interval and validity interval. The polynomial function is given by the standard form:

$$p(t) = a_1 t^n + a_2 t^{n-1} + \dots + a_n t + a_{n+1} \quad (1)$$

where $p(t)$ is the function evaluated at value t , in our case $p(t)$ is the Cartesian coordinate X , Y , or Z (or the satellite clock correction) and t is time—number of seconds from the beginning of the fit interval. The fit interval is the range of data (coordinates for the epochs within the range) used for the estimation of coefficients a_i , $i = 1, \dots, n + 1$. The fit interval must include at least $n + 1$ epochs to be able to fit an n -order polynomial. The concept of fit and validity interval is depicted in Fig. 2.

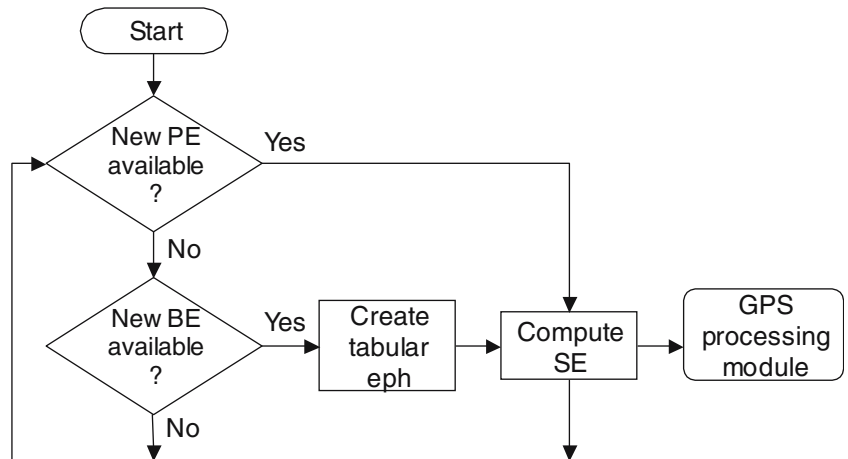
In this example, the fit interval is 3 h and contains 13 epochs, for which the satellite coordinates are listed in the tabular orbits. Therefore, the maximum order of polynomial that can be fitted to this interval is 12.

The estimation procedure of the a_i coefficients is done with the Matlab function *polyfit*, which estimates the coefficients in a least squares sense. To improve the numerical precision, the dataset p is normalized by centering it at a zero mean; by subtracting its mean value \bar{p} , and scaling it to a unit standard deviation by dividing each observation with the standard deviation σ_p as follows:

$$p_C = \left(\frac{p_i - \bar{p}}{\sigma_p} \right). \quad (2)$$

When interpolating a dataset using polynomial fitting, a problem known as Runge's phenomenon (RP) can occur (Dahlquist and Björck 1974). It arises when a higher degree polynomial is fit to equidistant data, like tabular orbits, and results in oscillations towards the end of the fit interval. Instead of trying to reduce the influence of RP by applying different interpolation methods, we solve the problem by ignoring the affected parts of the interval by adjusting the size of the validity interval. Therefore, the user has the possibility to choose a validity interval, which is always placed in the middle of the fit interval, see Fig. 2. The polynomial coefficients estimated from fit interval data are then used for coor-

Fig. 1 Flow diagram of the orbit module for real-time applications. *PE* precise ephemeris, *BE* broadcast ephemeris, *SE* standard ephemeris



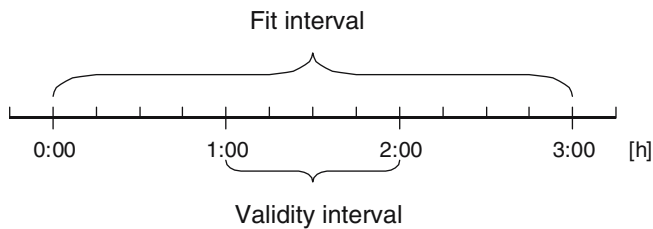


Fig. 2 Fit and validity interval

dinate computation only for epochs within the validity interval. The first fit interval begins at the first epoch in the tabular orbits; the next one begins n hours later, where n is the length of the validity interval.

We have created a set of Matlab functions based on an object-orientated approach. Full information on how to use these functions will be found in the document `Matlab_implementation.doc`, available at the GPS Toolbox website (<http://www.ngs.noaa.gov/gps-toolbox>). All of the Matlab m-files are available at this website also.

Tests

The following paragraphs outline several tests of our interpolation algorithm using real broadcast and precise ephemerides. Based on these tests, we try to find the most suitable values for the order of the polynomial, fit interval, and validity interval. For the following tests, we used the precise ephemeris file `igs13036.sp3` and the broadcast ephemeris file `auto0010.05n`, which were downloaded from SOPAC/CSRC archive (<http://lox.ucsd.edu/>). These files contain data for all GPS satellites during day 2005-01-01. All calculations and analyses were done with the data for satellite PRN01.

Runge's phenomenon

First, let us analyze the effect of RP on the interpolation with different: orders of polynomial, fit intervals, and validity intervals. For this purpose, we use parameters from the broadcast ephemeris, to be able to compute "true" reference coordinates for any epoch. Broadcast ephemerides are usually updated once every 2 h to maintain the specified accuracy level. The parameter updates result in small jumps in the computed satellite trajectories. To avoid the jumps during these tests, we use only one set of broadcast parameters for computation of tabular and then standard orbits. The standard orbits are used to interpolate coordinates for given epochs. These coordinates are then compared with coordinates calculated directly using the standard algorithm for the broadcast ephemeris (ARINC 2000). The results from

such a computation are shown in Fig. 3. The broadcast ephemeris parameters have their reference epoch at 04:00 and the standard orbits are based on a 12-coefficient polynomial with a 3 h fit interval.

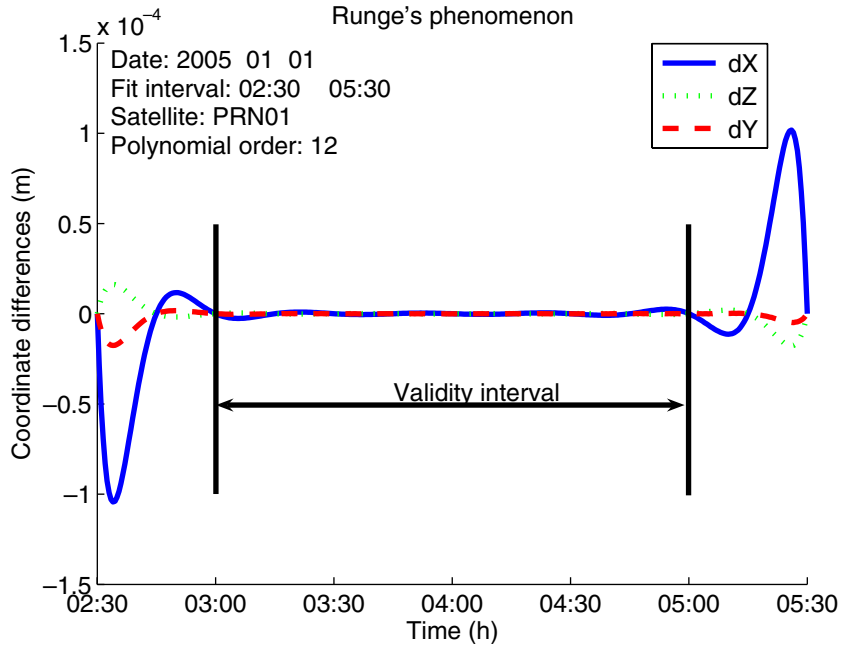
As seen in Fig. 3, RP influences mainly the first and last 30 min of the fit interval. This similar behavior can be also observed for several different sizes of fit intervals. RP will also affect the area inside the validity interval, but with significantly smaller oscillations. Some tests have been done to study the size of the oscillations for different fit intervals and different polynomial orders. The maximal oscillations have been computed for 3, 4, and 5 h fit intervals. The sizes of the oscillations get smaller with the increasing order of the polynomial. When the maximum number of coefficients were used (12, 16, and 20), for fit intervals of 2, 3, and 4 h, we achieved the following results: 0.025 mm, 10 μ m, and < 10 μ m, respectively. The results were already negligibly small (with respect to the coordinates precision) for polynomial order 10.

Broadcast ephemeris

Broadcast ephemeris parameters have a prescribed "validity" period, which are approximately 4 h (Hoffman-Wellenhof et al. 2001), 2 h before and 2 h after the reference epoch. This limits the maximum time for determining tabular orbits from a broadcast ephemeris, in real-time applications, to 4 h. As mentioned above, the broadcast ephemeris parameters are updated once every 2 h. It means that during a 4-h fit interval, there will be at least two parameter updates. Each time the broadcast ephemeris is updated a small jump will occur in the computed satellite trajectories, caused by the change of parameters. These jumps will influence the interpolation precision by additional oscillations within the validity interval. Figure 4 shows the case when the parameters are updated in the middle of the validity interval.

So far, we have compared coordinates computed from our standard ephemeris with coordinates calculated directly from the broadcast ephemeris. These differences show the behavior of the interpolation algorithm but nothing about the accuracy of the coordinates. To study the accuracy of interpolated and computed coordinates derived from broadcast ephemeris, we use coordinates derived from a precise ephemeris as "true" values. Figure 5 shows the coordinate differences between broadcast and precise ephemerides. The parameter shifts in the broadcast ephemeris occurring at odd hours and this can be seen in the form of sudden jump in the coordinate differences. The jumps are smoothed when using interpolated coordinates (thin lines). The difference in the accuracy of the interpolated versus the directly computed broadcast ephemeris is negligible.

Fig. 3 Differences between “true” and interpolated coordinates (broadcast ephemeris, one set of parameters)



Precise ephemeris

Precise ephemerides in SP3 format are tabular orbits, where satellite coordinates are usually given at 15-min intervals. Since we do not have the possibility to get “true” coordinates within the 15-min interval, we test the interpolation precision by omitting one epoch

when computing our standard ephemeris. In this way, we can compare the interpolated coordinates with the “true” ones, which were not used for polynomial fitting. The results from such a test are rather pessimistic, since we interpolate over a 30-min interval instead of the usual 15-min interval. To be consistent with the tests done with the broadcast ephemeris, we use a 4-h fit interval in the following tests. Figure 6

Fig. 4 Differences between interpolated and computed satellite coordinates (broadcast ephemeris, two sets of parameters, shift at 05:00)

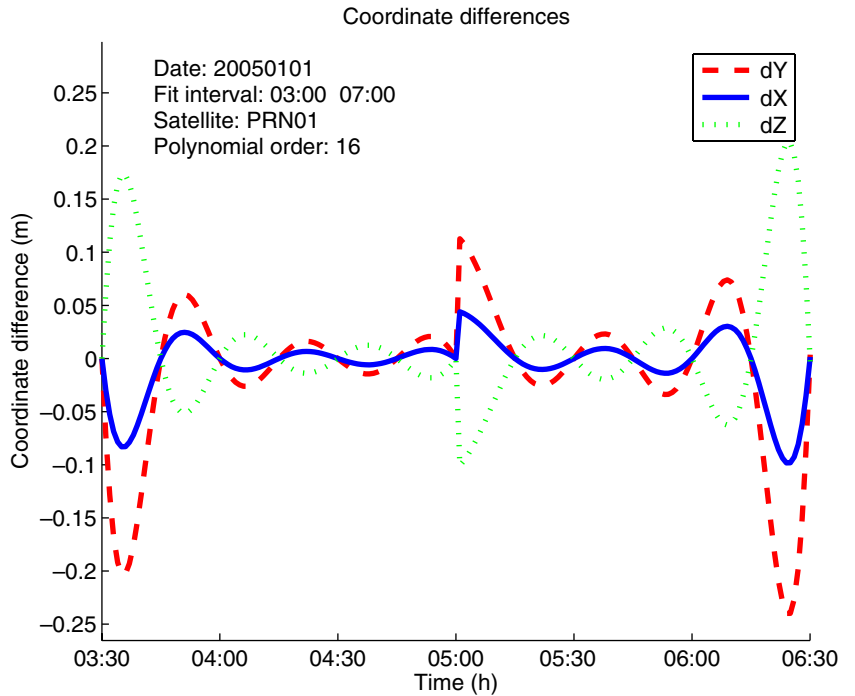
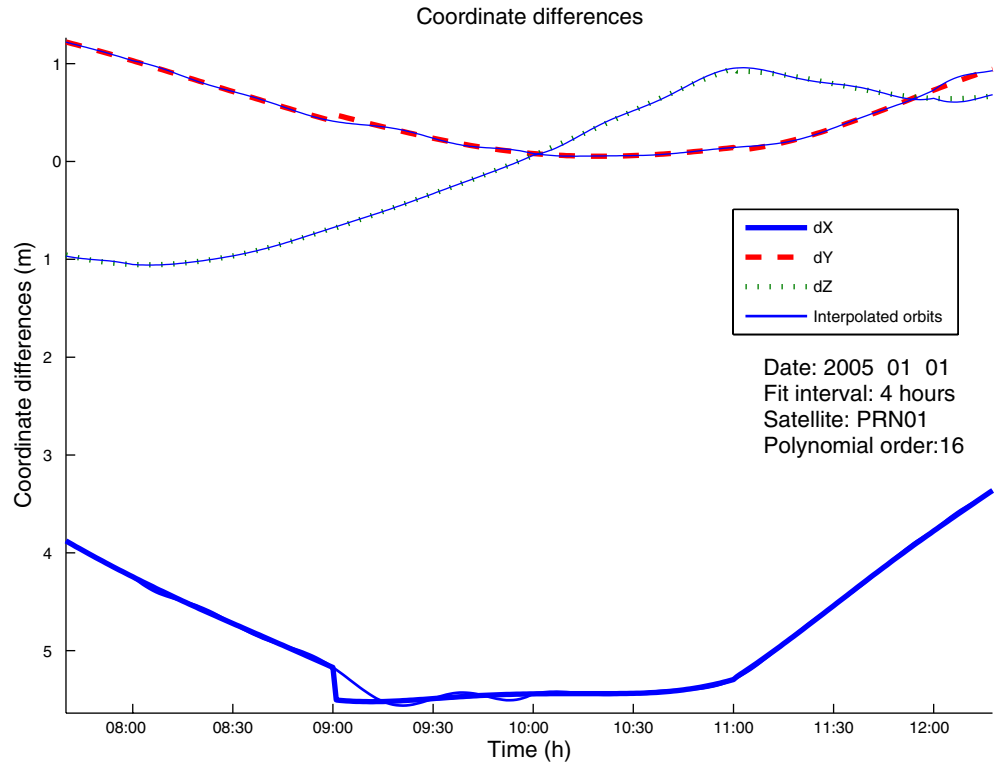


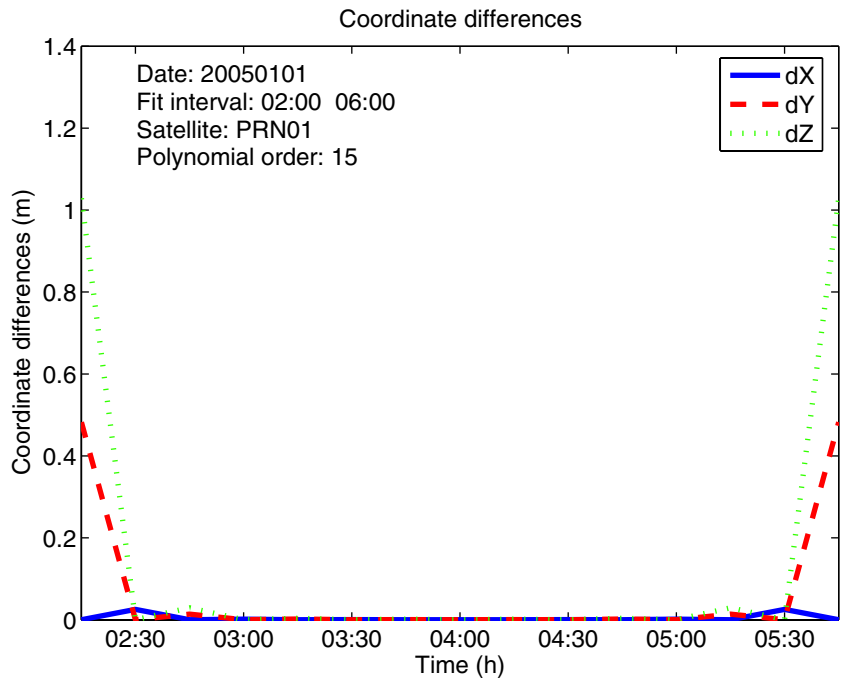
Fig. 5 Differences between the broadcast (both computed and interpolated) and the precise ephemeris



shows differences between interpolated and “true” coordinates, where the epochs were removed sequentially within the fit interval. The differences are large if we omit the epochs at the beginning and at the end of fit interval, but they are sufficiently small in the middle of the fit interval.

The maximum difference for a 2-h validity interval is 4 mm, and for a 1-h validity interval, 2 mm. These differences are far below the precision level of the precise ephemeris. Furthermore, we expect that they will get even smaller in real applications, when we interpolate at 15-min intervals.

Fig. 6 Absolute value of the coordinate differences between true tabular and interpolated coordinates (30-min epochs, precise ephemeris)



Conclusions

We have implemented and tested an interpolation method for GPS satellite orbits. The implementation allows the user to choose several interpolation parameters. Based on tests with broadcast and precise ephemerides, we recommend the following values: 4-h fit interval, 2-h validity interval, and polynomial order 16. The interpolation error is at the millimeter level when using these settings. The interpolation precision

decreases outside of the validity interval. Therefore, in cases where one needs satellite coordinates close to the end of fit interval, we recommend adding new ephemeris data and computing a new standard ephemeris.

Acknowledgments We are thankful to the Swedish Research Council for Environment, Agricultural Sciences and Spatial Planning, FORMAS, for sponsoring this research within the framework of "Monitoring of construction and detection of movements by GPS ref no. 2002-1257".

References

- ARINC (2000) Interface Control Document ICD-GPS-200C, IRN-200C-004, April 12, ARINC Research Corporation, El Segundo, CA. Available at <http://www.navcen.uscg.org/pubs/gps/icd200/icd200cw1234.pdf>
- Dahlquist G, Björck Å (1974) Numerical methods. Prentice-Hall, Englewood Cliffs
- Hoffman-Wellenhof B, Lichtenegger H, Collins J (2001) GPS theory and practice, 5th edn. Springer, Berlin Heidelberg New York, p. 67. ISBN 3-211-83534-2
- Hugentobler U, Schaer S, Fridez P (2001) Documentation of the Bernese GPS Software, Version 4.2. University of Bern, Bern, Switzerland
- Schenewerk M (2003) A brief review of basic GPS orbit interpolation strategies. GPS Solut 6(4):265–267