Baselga Moreno, S. (2014). Ambiguity-Free Method for Fast and Precise GNSS Differential Positioning. Journal of Surveying Engineering. 140(1):22-27. doi:10.1061/(ASCE)SU.1943-5428.0000111.

Ambiguity-Free Method for Fast and Precise GNSS Differential Positioning

Sergio Baselga, M.ASCE[1]

Abstract: Methods based on integer ambiguity determination, such as the least squares ambiguity decorrelation adjustment (LAMBDA) method, are currently used for precise global navigation satellite systems (GNSS) differential positioning. In the present paper we propose an ambiguity-free method based on a dedicated mixed (stochastic/deterministic) optimization algorithm that unlike the LAMBDA method is capable of providing reliable and accurate results using few observation epochs (e.g. 1-cm accuracy with just two epochs) having the additional advantages of insensitiveness to cycle slips and impossibility of wrong ambiguity fixation. In addition, we will demonstrate that the application of the linear (deterministic) part of our algorithm yields the correct baseline results much easier and faster than methods requiring integer ambiguity determination provided the initial approximate coordinates are accurate to a few centimeters. However, the use of ambiguity-free methods requires that the integer character of the ambiguities be preserved so that they can be eliminated; therefore no ionosphere-free combination can be computed and the methods are valid only for short baselines (e.g. less than 10 km).

CE Database subject headings: Global positioning systems; Optimization; Measurement.

Introduction

GNSS differential positioning in short baselines has the advantage of effective cancellation of common error sources. Precise GNSS positioning needs the use of carrier phase observables, which inevitably leads to the appearance of the integer-valued unknowns called ambiguities. Since the first days of GNSS positioning extensive research has been dedicated to the optimization of the process by which ambiguities are initially approximated, fixed to integer

values, the corresponding solution obtained – i.e. the real-valued coordinates and integer-valued ambiguities – and the solution validated (Hatch 1990, Frei and Beutler 1990, Chen 1993, Teunissen 1993, 1995; Kim and Langley 2000, Verhagen 2004, Li and Shen 2010, as examples in the abundant literature).

There are also other methods based on eliminating ambiguities from the equations exploiting their integer character instead of obtaining their particular values (Counselman and Gourevitch 1981, Remondi 1991, Mader 1992, Wang *et al* 2007, Cellmer *et al* 2010, Cellmer 2012). In theory, these methods have some clear advantages: they are unaffected by cycle slips, wrong ambiguity fixation is not possible – since no ambiguity estimation is done –, and they have much less unknowns to determine (three, instead of three plus all the integer ambiguities). However, their need for initial good approximate coordinates is a drawback that has prevented a higher degree of usefulness. This was remedied in Baselga (2010) by using an optimization procedure insensitive to the degree of accuracy of the initial coordinates. Its computational cost, however, still made it little competitive with the standard methods based on ambiguity determination (for example the popular LAMBDA method).

In the present paper a more successful proposal will be developed: in the next section an ambiguity-free functional model will be derived and further solved by a dedicated mixed (stochastic/deterministic) optimization method. Beyond the obvious advantages of insensitiveness to cycle slips and impossibility of obtaining a wrong solution due to a wrong ambiguity fixation, the application section will show how the proposed procedure clearly overcomes the renowned LAMBDA method for short time-span baselines.

In addition, it will also become evident that for a baseline with good approximate coordinates one does not need to consider ambiguities at all, since our ambiguity-free linear model let us

[1] Dpto. Ingeniería Cartográfica, Geodesia y Fotogrametría, Universidad Politécnica de Valencia, Camino de Vera s/n, 46022 Valencia, Spain. Email: serbamo@cgf.upv.es.

obtain the correct solution reliably and much faster than the standard methods based on ambiguity determination.

Functional model and proposed algorithm

Let us write the double-differenced carrier phase observation equation as:

$$\lambda\phi_{ij}^{kl} = \rho_{ij}^{kl} + \lambda N_{ij}^{kl} + \varepsilon_{ij}^{kl} \tag{1}$$

where λ is the carrier wavelength, ϕ_{ij}^{kl}, ρ_{ij}^{kl} and N_{ij}^{kl} are respectively the double-differenced carrier phase observation, range and integer ambiguity, for subtractions with the following order

$$\phi_{ij}^{kl} = \left(\phi_j^l - \phi_i^l\right) - \left(\phi_j^k - \phi_i^k\right) = \phi_j^l - \phi_i^l - \phi_j^k + \phi_i^k \tag{2}$$

where subscripts denote baseline stations *i* and *j* and superscripts denote satellites *l* and *k*. Finally, for a sufficiently short baseline ε_{ij}^{kl} is a zero-centered with standard deviation σ Gaussian residual, expectably a few millimeters for a typical double-differenced carrier phase equation.

Let us assume coordinates of station *i* to be known (more rigorously, let us say we assume them to be known for the mean epoch of observation in the ephemerides coordinate frame within, say, a few millimeters) and let us use some approximate coordinates for station *j* – X_{j0}, Y_{j0}, Z_{j0} – therefore we can expand ρ_{ij}^{kl} around them and write

$$\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl} = \left(\frac{\partial\rho_{ij}^{kl}}{\partial X_j}\right)_0 dX_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Y_j}\right)_0 dY_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Z_j}\right)_0 dZ_j + \lambda N_{ij}^{kl} + \varepsilon_{ij}^{kl} \tag{3}$$

where the subscript *0* denotes the particular value obtained after using the approximate coordinates for station *j* and dX_j, dY_j, dZ_j are the corrections to the approximate coordinates (to be determined in the estimation process).

Let us now examine the case where the approximate *j*-coordinates are quite accurate (we will explicitly obtain the required degree of accuracy in a later section, for the moment we can think of 1 cm just to have a figure in mind). Dividing Eq. (3) by λ we obtain

$$\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda} = \left(\frac{\partial\rho_{ij}^{kl}}{\partial X_j}\right)_0 \frac{1}{\lambda} dX_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Y_j}\right)_0 \frac{1}{\lambda} dY_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Z_j}\right)_0 \frac{1}{\lambda} dZ_j + N_{ij}^{kl} + \frac{1}{\lambda}\varepsilon_{ij}^{kl} \tag{4}$$

If, as said, the approximate *j* coordinates are quite accurate, the first three terms on the right side – and also the last one – are all much smaller than 1. Therefore the result on the left side will read something like e.g. 85.018 and we can assure the non-integer part 0.018 to be the sole contribution of the dX_j, dY_j, dZ_j and ε_{ij}^{kl} summands, whereas the integer part 85 has to be the N_{ij}^{kl} term.

This leads to a rather straightforward ambiguity-free method though not useful unless we have accurate enough approximate coordinates:

$$\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda} - Int\left(\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda}\right) = \left(\frac{\partial\rho_{ij}^{kl}}{\partial X_j}\right)_0 \frac{1}{\lambda} dX_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Y_j}\right)_0 \frac{1}{\lambda} dY_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Z_j}\right)_0 \frac{1}{\lambda} dZ_j + \frac{1}{\lambda}\varepsilon_{ij}^{kl} \tag{5}$$

Int denoting the rounding to next integer function. After reordering summands, the corresponding set of equations in matrix form reads

$$\boldsymbol{Ax} = \boldsymbol{l} + \boldsymbol{v} \tag{6}$$

with coefficient matrix $\boldsymbol{A}$ and vector of unknowns $\boldsymbol{x}$, observations $\boldsymbol{l}$ and residuals $\boldsymbol{v}$, which can be easily solved by least squares.

Evidently, having got rid of ambiguities in Eq. (5) has multiple advantages: the functional model is insensitive to possible cycle slips, incorrect ambiguity fixation and validation are not an issue now, and matrix formation and manipulation are much faster (since there are only three unknowns: dX_j, dY_j, dZ_j). The only drawback is the need for accurate enough approximate coordinates.

Now imagine the approximate *j*-coordinates were not only accurate but *exact*, then $dX_j = dY_j = dZ_j = 0$ in Eq. (5) and we would obtain for this fortunate occasion

$$\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda} - Int\left(\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda}\right) = \frac{1}{\lambda}\varepsilon_{ij}^{kl} \tag{7}$$

The interesting thing here is that these coordinates *do exist*, that is, there are coordinates for station *j* that make Eq. (7) hold with ε_{ij}^{kl} being purely Gaussian zero-centered residuals with standard deviation of a few millimeters. Our only task is finding them!

We could try some brute-force approach consisting in iteratively plugging random *j*-coordinates into Eq. (7) and evaluating the corresponding residuals. More properly, we should understand this as an optimization problem

$$\begin{cases} min\ f \\ \text{being } f(X_j, Y_j, Z_j) = \sum\left(\frac{1}{\lambda}\varepsilon_{ij}^{kl}\right)^2 = \sum\left[\frac{\lambda\phi_{ij}^{kl} - \rho_{ij}^{kl}}{\lambda} - Int\left(\frac{\lambda\phi_{ij}^{kl} - \rho_{ij}^{kl}}{\lambda}\right)\right]^2 \\ \text{for } X_j, Y_j, Z_j \end{cases} \tag{8}$$

and search for these coordinates by means of any global optimization method, e.g. simulated annealing (which, basically, puts some order into the random search). For our problem at hand, i.e. solving Eq. (8), this stochastic procedure is however very costly from the computational point of view.

Observing the features of the search space we can design now an adapted mixed – half stochastic, half deterministic – optimization algorithm that dramatically reduces the computational burden.

The search space can be symbolically depicted in Fig. 1.

Figure 1

It is a landscape composed of different attraction basins, pull-in regions (Verhagen 2004), or Voronoi cells (Cellmer 2012), having each of them a single local optimum. Each region

corresponds to a set of integer ambiguities, and among the points in a region (representing possible coordinate solutions) the local optimum is the best coordinate solution compatible with the set of ambiguities. There is a global optimum (the local optimum of a particular region): the correct solution for the baseline.

In our problem – Eq. (8) – the search space retains this underlying structure, even if we do not use ambiguities explicitly. Therefore, we can understand why a blind random search is so ineffective therein. Conversely, we can design a mixed approach: first, we randomly select a point in the search space, and, second, with these approximate coordinates we apply Eq. (5) to obtain either the global optimum if we have been so fortunate as to have fallen in the corresponding pull-in region – since Eq. (5) is rigorously applicable – or just an approximation (more or less coarse) to the local optimum in the corresponding cell. Then we evaluate the function *f* at the obtained point and take it as the origin for the next iteration, or not, depending on the function value.

In the appendix we give the pseudocode for a mixed algorithm using the simulated annealing method in the stochastic part, named part *a* (refer, e.g. to Baselga 2010 for details on the simulated annealing method) and the linear refinement Eq. (5) as the deterministic part (named part *b*) of every iteration. The solution evolution along iterations is shown in Fig. 2: the curved lines show the stochastic jumps – e.g. from step *0b* to *1a* – with statistically decreasing amplitudes (i.e. successive jumps are drawn from normal distributions having increasingly smaller standard deviations), while the straight segments – e.g. from *1a* to *1b* – show the deterministic refinement. The process converges to the optimum solution after not many iterations (depending them on the complexity of the problem), at any rate several degrees of order less than the sole application of simulated annealing, as in Baselga (2010).

Figure 2

As it can be demonstrated based on Cellmer (2012) work, the necessary condition for the proposed method to succeed in finding the global optimum is that the algorithm falls at least once in the correct Voronoi cell. For a search space of volume L^3 – with L the search space width – and assuming it to be made up of cubic Voronoi cells of diagonal λ – and consequently, volume $\left(\frac{\lambda}{\sqrt{3}}\right)^3$ – we have a number N of Voronoi cells

$$N = \frac{L^3}{\lambda^3} 3^{3/2} \tag{9}$$

If the algorithm consisted in visiting one after another all the cells in the search space then we would need N iterations to ensure success. Considering that the simulated annealing method has to considerably improve on that exhaustive procedure, but also that N in Eq. (9) is only a rough and low estimate due to the naive discretization we have made of the search space (e.g. any other type of parallelepiped-shaped cell of diagonal λ would lead always to larger N values) we can take N as a conservative estimate for the required number of iterations in our algorithm. The experiments conducted in the following section corroborated that for design parameters – see appendix A – leading to some N or $N/2$ iterations succesful determination always occurred, whereas values of some $N/3$ or lower provided occassionally suboptimal solutions.

Application

BAY5 and BAY6 are two GPS stations separated some 29.98 m belonging to CORS network (Snay and Soler, 2008). They are taken as a reliable source of data for applying the before mentioned differential positioning method: they have stable monumentation, regular observations every 30 s and, in this case, even the same antennas and radomes.

We will consider – for instance – a sample of consecutive observations starting on January 14, 2011, 22:00 (GPS time). As said, we are aimed at showing the estimation abilities of the

proposed procedures compared with the standard approach of ambiguity determination, in particular with the LAMBDA method. The software used has been developed under Matlab™ by the author in what refers to generalities and the ambiguity-free methods, and by Borre (2009) the part of the LAMBDA method.

In all the computations below we first process code observables in order to obtain the approximate coordinates to be used in all types of carrier phase processing: the LAMBDA method and two ambiguity-free methods: the (general) mixed optimization procedure in Eq. (8) and the straightforward linear procedure in Eq. (5) only devised for good approximate coordinates. Due to the short baseline length only the use of L_1 carrier phases is advisable.

With one hour of observations (121 epochs) the three solutions are coincident within 1 mm. We will take the solution obtained by the LAMBDA method as the reference solution for the rest of all computations (shown in Table 1). Computation times appearing in the table refer to a PC using an Intel Core2 Quad CPU at 2.33 GHz with 2.96GB RAM and are given for the sole purpose of providing a rough relative comparison among methods. Even so, we make a note of caution regarding the general ambiguity-free procedure (Eq. 8): times are strongly dependent on the β value used in the algorithm. Here we have used different β values: values quite close to unity (like β = 0.999) for assuring a correct solution with short observation spans (a few minutes) where the solution is “not much clear”, and faster values (like β = 0.9) for the longest observation spans. The usage of excessively close to unity values with large amount of data is too conservative and, correspondingly, makes computations unnecessarily costly. The usage of extremely relaxed or faster values (too far from 1) yields wrong solutions, their incorrectness being quite evident since the corresponding values for $\hat{\sigma}_v$ and $\hat{\mu}_v$ are far from the expected ones: typically a few millimeters for the double differences residuals, and no significant deviation from zero millimeters for the mean; it might also be pertinent to perform a statistical test for those hypotheses, although the values obtained in the

example are clearly reasonable. As a rule of thumb, the user may resort to the previous Eq. (9) in order to obtain a reasonable number of iterations to be done.

It can be seen that the general ambiguity-free procedure – Eq. (8) – performs successfully, at worse, with computational times around 1 s.

Table 1

If approximate coordinates are good enough (all below 6 cm in this example), which is achieved for at least 45 min of observations, all three methods perform successfully, with the linear ambiguity free method clearly faster than LAMBDA and general ambiguity-free methods.

For observation times ranging from 5 to 40 min the approximate coordinates computed by processing code observations are not so good. Therefore the linear ambiguity free method fails as expected, while the LAMBDA and the general ambiguity-free method are successful. It has been checked – though not included in the table – that if we use for these time spans good approximate coordinates – all below 6 cm as before – the linear ambiguity free method gives results compatible with those of LAMBDA and the general ambiguity-free method albeit much faster (the times being similar to those appearing now in the table for the incorrect results).

Finally, it is shown that for short observation times, four minutes and below, and approximate coordinates of a bad quality, the LAMBDA method fails while the general ambiguity-free method is still able to retain millimetre accuracy or 1.0 cm at worse for time spans as short as 2 epochs (21:00:00 to 21:00:30).

No meaningful results were obtained for just a single epoch (the LAMBDA method was even underdetermined).

Coda: with good coordinates never mind ambiguities (solution is straightforward!)

There are occasions where the initial approximate coordinates are known within a few centimeters, e.g. we can imagine a surveying network for deformation monitoring.

For those cases, all methods based on ambiguity determination have inexcusably to be abandoned in favor of the linear ambiguity-free method given in Eq. (5) which not only has the before mentioned advantages of insensitiveness to cycle slips and impossibility of wrong ambiguity fixation, but it is also much simpler and faster. Going back again to Fig. 2 our problem now reduces to simply going from *7a* to *7b*, since we are in the correct pull-in region from the very beginning.

As it was said before, it has been experienced (results shown in Table 1, and also others not displayed) that the linear ambiguity-free method succeeds for the preceding baseline if the approximate coordinates are more accurate than some 6 cm.

Now, we want to rigorously obtain the bound for the correct application – even in the worst cases – of this linear ambiguity-free method.

Let us go back to Eq. (5) and analyze the first doubtful case, i.e. when the fraction $\frac{\lambda\phi_{ij}^{kl} - \rho_{ij\ 0}^{kl}}{\lambda}$ is halfway between two integers; then

$$0.5 = \left(\frac{\partial\rho_{ij}^{kl}}{\partial X_j}\right)_0 \frac{1}{\lambda} dX_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Y_j}\right)_0 \frac{1}{\lambda} dY_j + \left(\frac{\partial\rho_{ij}^{kl}}{\partial Z_j}\right)_0 \frac{1}{\lambda} dZ_j + \frac{1}{\lambda}\varepsilon_{ij}^{kl} \tag{10}$$

Since dX_j, dY_j, dZ_j are of the order centimeters and ε_{ij}^{kl} a few millimeters it can be neglected with no significant error. Developing partial derivatives we can write

$$0.5\lambda = \left(\frac{X_s - X_{j0}}{\rho_{j0}^s} - \frac{X_r - X_{j0}}{\rho_{j0}^r}\right) dX_j + \left(\frac{Y_s - Y_{j0}}{\rho_{j0}^s} - \frac{Y_r - Y_{j0}}{\rho_{j0}^r}\right) dY_j + \left(\frac{Z_s - Z_{j0}}{\rho_{j0}^s} - \frac{Z_r - Z_{j0}}{\rho_{j0}^r}\right) dZ_j \tag{11}$$

Simple variable changes let us write

$$0.5\lambda = (x_s - x_r) dX_j + (y_s - y_r) dY_j + (z_s - z_r) dZ_j \tag{12}$$

where $\sqrt{x_r^{\ 2} + y_r^{\ 2} + z_r^{\ 2}} = 1$ and $\sqrt{x_s^{\ 2} + y_s^{\ 2} + z_s^{\ 2}} = 1$.

We are interested in the least coordinate deviation from its exact value, say δ, that makes the solution to be doubtful. For a given δ, it is clear that the worst case is when all coordinates are this far from the exact solution $|dX_j| = |dY_j| = |dZ_j| = \delta$. Without loss of generality for assuming positive signs $dX_j = dY_j = dZ_j = \delta$ we can rewrite Eq. (12) as

$$0.5\lambda = (x_s - x_r + y_s - y_r + z_s - z_r)\delta \tag{13}$$

and obtain

$$\delta = \frac{0.5\lambda}{x_s + y_s + z_s - (x_r + y_r + z_r)} \tag{14}$$

For δ to be minimum in Eq. (14) – since we are interested in the least δ that makes the equation to be doubtful – we need the denominator to be maximum. A little calculus shows that this happens for $x_s = y_s = z_s = \frac{1}{\sqrt{3}}$ and $x_r = y_r = z_r = -\frac{1}{\sqrt{3}}$ (reversing signs is also possible but note it is also possible for the doubtful numerator to be -0.5λ). Upon substitution in Eq. (14) we obtain the worst case value

$$\delta = \frac{\lambda}{4\sqrt{3}} \tag{15}$$

This represents 2.75 cm for L_1 carrier phase observations. Evidently this is a too conservative estimate (we succeeded in the previous example with coordinates approximated to 5 cm): first, because in practice not all coordinates are inaccurate by the same amount – an assumption that brought our Eq. (12) into Eq. (13) – and second because two visible satellites cannot be diametrically opposed (with one of them just above the horizon we would have the other in the opposite direction slightly below the horizon), which is implied by our worst-case solution to Eq. (14). All in all we have determined a conservative bound for the approximate coordinate accuracy below which the linear ambiguity-free method given in Eq. (5) is always successful.

Conclusions

Methods based on ambiguity fixing, such as the LAMBDA method, deal with a complex search space. Not only there are more unknowns than the three coordinates, but also those extra unknowns have to be reliably fixed to integers. Therefore, the use of a method based on ambiguity determination has the certain drawback of high computational cost, plus the possible problems derived from its sensitiveness to cycle slips and possible wrong ambiguity fixation, which is a critical issue.

The general ambiguity-free method derived in this paper, in addition to its insensitiveness to cycle slips and impossible wrong ambiguity fixation, allows for an accurate and reliable baseline determination with the use of much less epochs than the LAMBDA method: for instance, in the example baseline we obtained a 1 cm accuracy with only two epochs.

In addition, for the particular case of initial approximate coordinates with accuracies of a few centimeters, we devised a reliable linear ambiguity-free method much faster than LAMBDA and exempt from their drawbacks.

Acknowledgements

The author is grateful to the editor and the anonymous reviewers for their valuable suggestions, corrections and comments that helped improve the original manuscript.

References

Baselga, S. (2010) “Global optimization applied to GPS positioning by ambiguity functions.” *Meas. Sci. Technol.*, 21(12), 125102(5pp).

Borre, K. (2009) “GPS EASY Suite II. A Matlab companion.” *InsideGNSS*, 4(2): 48-52.

Cellmer, S. (2012) “A graphic representation of the necessary condition for the MAFA method.” *IEEE Trans. Geosci. Remote Sensing*, 50(2): 482-488.

291 Cellmer, S., Wielgosz, P., and Rzepecka, Z. (2010) “Modified ambiguity function approach
292 for GPS carrier phase positioning.” *J. Geod.*, 84(4), 267-275.
293 Chen, D. (1993) “Fast Ambiguity Search Filter (FASF): a novel concept for GPS ambiguity
294 resolution.” *Proc., 6th Int. Technical Meeting of the Satellite Division of the U.S. Institute of*
295 *Navigation,* Salt Lake City, 781–787.
296 Counselman, C. C., and Gourevitch, S. A. (1981) “Miniature interferometer terminals for
297 Earth surveying: ambiguity and multipath with global positioning system.” *IEEE Trans.*
298 *Geosci. Remote Sens.*, GE-19(4), 244–252.
299 Frei, E., and Beutler, G. (1990) “Rapid static positioning based on the fast ambiguity
300 resolution approach “FARA”: theory and first results.” *Manuscr. Geod.*, 15(4), 325-356.
301 Hatch, R. (1990) “Instantaneous ambiguity resolution.” *Proc., IAG Int. Symp. 107 on*
302 *Kinematic Systems in Geodesy, Surveying and Sensing*, Banff, Canada, 299-308.
303 Kim, D., and Langley, R. B. (2000) “GPS ambiguity resolution and validation:
304 methodologies, trends and issues.” *Proc., 7th GNSS Workshop-Int. Symp. GPS/GNSS*, Seoul,
305 Korea, 213-221.
306 Li, B., and Shen, Y. (2010) “Global Navigation Satellite System ambiguity resolution with
307 constraints from normal equations.” *J. Surv. Eng.*, 136(2), 63-71.
308 Mader, G. L. (1992) “Rapid static and kinematic global positioning system solutions using the
309 ambiguity function technique.” *J. Geophys. Res.*, 97(B3), 3271–3283.
310 Remondi, B. W. (1991) “Pseudo-kinematic GPS results using the ambiguity function
311 method.” *Navigation*, 38, 17-36.
312 Snay, R. A., and Soler, T. (2008). “Continuously Operating Reference Station (CORS):
313 History, applications, and future enhancements.” *J. Surv. Eng.*, 134(4), 95-104.
314 Teunissen, P. J. G. (1993) “Least-squares estimation of the integer GPS ambiguities.” *IAG*
315 *General Meeting, Invited Lecture, Section IV Theory and Methodology*, Beijing, China.

316 Teunissen, P. J. G. (1995) “The least-squares ambiguity decorrelation adjustment: A method
317 for fast GPS integer ambiguity estimation.” *J. Geod.*, 70(1-2), 65-82.
318 Verhagen, S. (2004) “Integer ambiguity validation: an open problem?” *GPS Solut.*, 8(1), 36-
319 43.
320 Wang, Y. Q., Zhan, X. Q., and Zhang, Y. H. (2007) “Improved ambiguity function method
321 based on analytical resolution for GPS attitude determination.” *Meas. Sci. Technol.*, 18, 2985-
322 2990.

Appendix. Mixed optimization method pseudocode

Input parameters

Satellite coordinates X_k, Y_k, Z_k, X_l, Y_l, Z_l

Fixed station coordinates X_i, Y_i, Z_i

Search space (wide as desired): [$X_{j\text{-}min}$, $X_{j\text{-}max}$], [$Y_{j\text{-}min}$, $Y_{j\text{-}max}$], [$Z_{j\text{-}min}$, $Z_{j\text{-}max}$]

Double-differenced observations: ϕ_{ij}^{kl}

Carrier wavelength: λ

(optional) Approximate station coordinates X_j, Y_j, Z_j. It is sensible to use them at least for defining the search space boundaries as $X_j \pm \Delta$, $Y_j \pm \Delta$, $Z_j \pm \Delta$

Design parameters

Initial standard jump amplitude (advisably 1/10 to 1/3 of search space width): σ_o

Final standard jump amplitude (for the global optimum to have been reached): σ_f = 0.03 m

Coefficient for exponential decay of standard jump amplitudes: β (e.g. β = 0.999)

Probability of accepting worse solutions (small, helps avoid local minima): p (e.g. p = 0.01)

Pseudocode algorithm

// Iteration 0

// Part *a* (stochastic) (only if we do not have initial approximate coordinates)

Select any coordinates $(X_j, Y_j, Z_j)_{0\_a}$ belonging to the search space (even at random)

// Part *b* (deterministic)

Compose matrix A following Eq. (5)-(6) using $(X_j, Y_j, Z_j)_{0\_a}$

Compose vector l following Eq. (5)-(6) using $(X_j, Y_j, Z_j)_{0\_a}$

Solve the least squares problem in Eq. (6) to obtain $(dX_j, dY_j, dZ_j)_0$

// Refine coordinates:

$(X_j, Y_j, Z_j)_{0\_b} = (X_j, Y_j, Z_j)_{0\_a} + (dX_j, dY_j, dZ_j)_0$

f_0 = Evaluate cost function f by using $(X_j, Y_j, Z_j)_{0\_b}$ in Eq. (8)

$f_{best} = f_0$

$x_{best} = (X_j, Y_j, Z_j)_{0\_b}$

$\sigma_1 = \beta\sigma_o$ // standard jump amplitude is reduced

// Iteration t

while $\sigma_t < \sigma_f$ // until the standard jump amplitude reaches the final size

// Part *a* (stochastic)

Generate Gaussian random displacements $\Delta X_t \sim N(0, \sigma_t)$, $\Delta Y_t \sim N(0, \sigma_t)$, $\Delta Z_t \sim N(0, \sigma_t)$

// Obtain new coordinates by displacing the ones in the last iteration:

$(X_j, Y_j, Z_j)_{t\_a} = (X_j, Y_j, Z_j)_{t\text{-}1\_b} + (\Delta X_t, \Delta Y_t, \Delta Z_t)$

// Part *b* (deterministic)

Compose matrix A following Eq. (5)-(6) using $(X_j, Y_j, Z_j)_{t\_a}$

Compose vector l following Eq. (5)-(6) using $(X_j, Y_j, Z_j)_{t\_a}$

Solve the least squares problem in Eq. (6) to obtain $(dX_j, dY_j, dZ_j)_t$
// Refine coordinates:
$(X_j, Y_j, Z_j)_{t\_b} = (X_j, Y_j, Z_j)_{t\_a} + (dX_j, dY_j, dZ_j)_t$
f_t = Evaluate cost function f by using $(X_j, Y_j, Z_j)_{t\_b}$ in Eq. (8)
// Accept or discard as origin for the next iteration jump
if $f_t < f_{t-1}$
    $(X_j, Y_j, Z_j)_{t\_b} = (X_j, Y_j, Z_j)_{t\_b}$ // i.e. do nothing but retain this as origin for next iteration
    if $f_t < f_{best}$
        $f_{best} = f_t$
        $x_{best} = x_t$
    end
else
    if *random_number_from_uniform_distribution_[0,1]* < p
        $(X_j, Y_j, Z_j)_{t\_b} = (X_j, Y_j, Z_j)_{t\_b}$ // i.e. accept worse solutions with low probab. p
    else
        $(X_j, Y_j, Z_j)_{t\_b} = (X_j, Y_j, Z_j)_{t-1\_b}$ // do not accept as origin for next iteration
    end
end
$\sigma_{t+1} = \beta\sigma_t$ // standard jump amplitude is reduced
end

// The solution is x_{best}, almost always coincident with x_t being t the last iteration
// The mean quadratic residual in Eq. (6) must be a few mm, the average residual statistically compatible with 0.
// Repeated executions must provide the same result (sufficient, though not necessary condition, of having
// attained the global optimum)

Figure captions

Fig. 1. Figurative representation of the search space.

Fig. 2. Course of iterations (part *a* stochastic, part *b* deterministic)

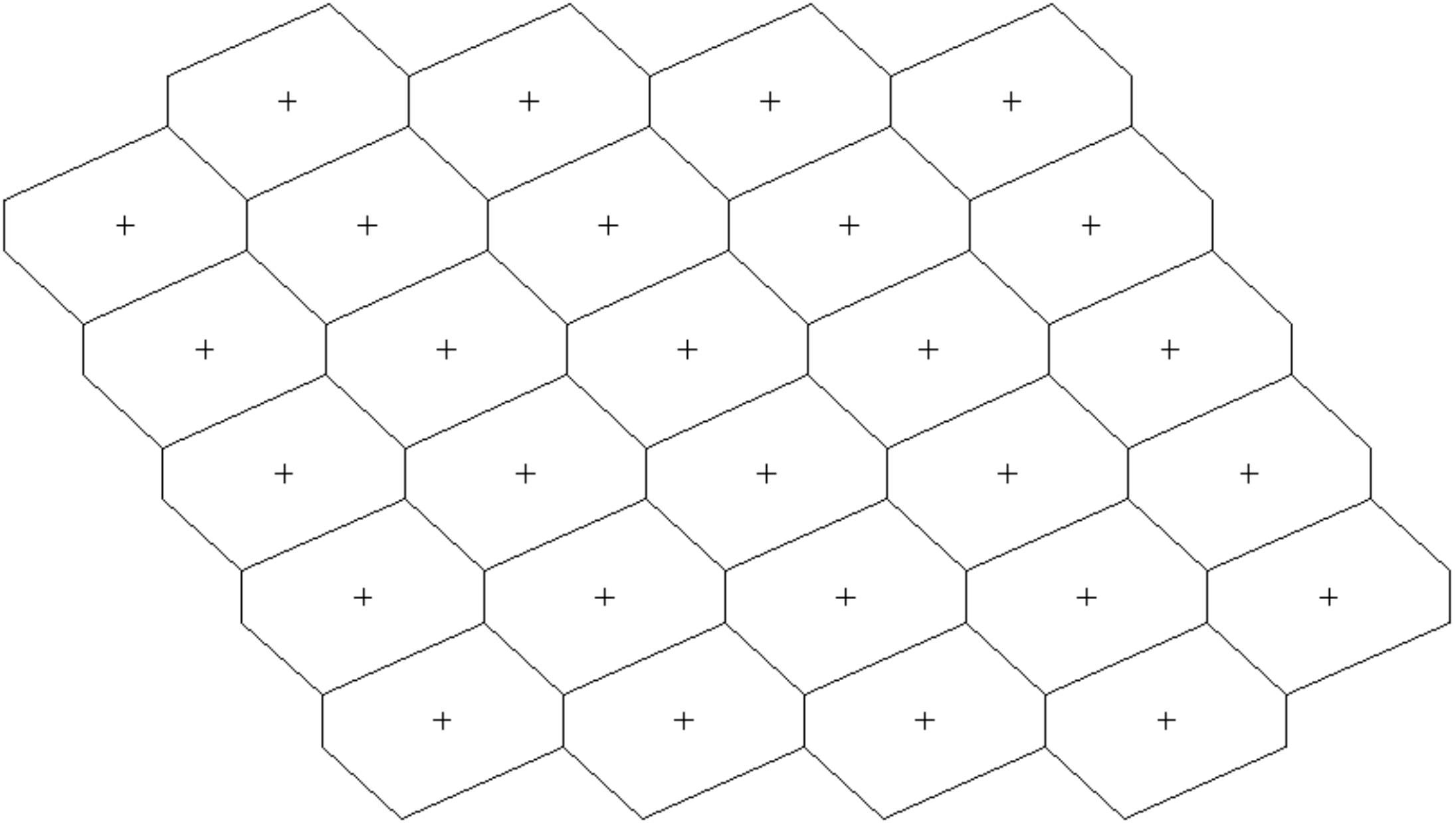
Tables

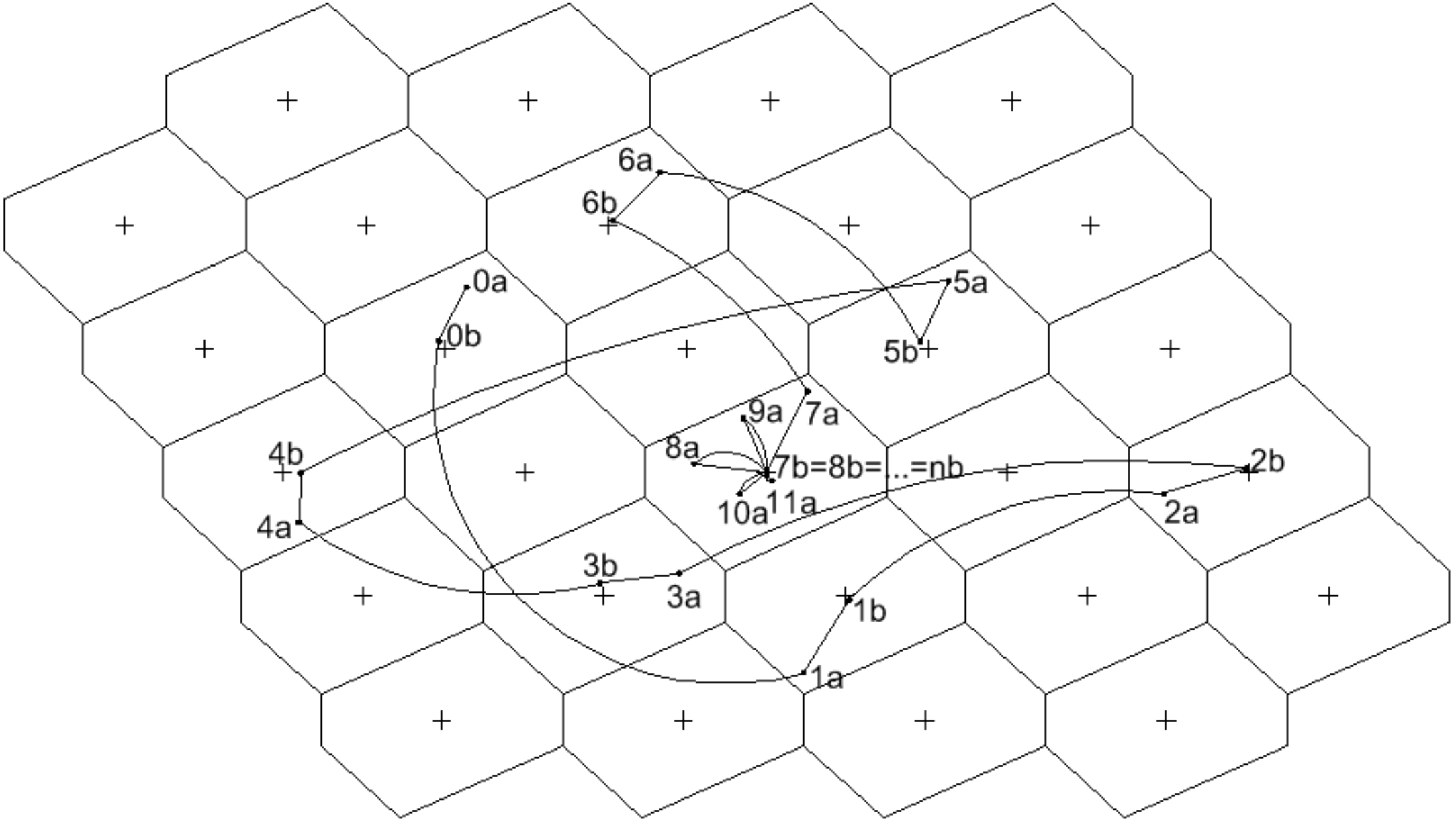
Table 1. Coordinate accuracy results using the LAMBDA method, the general Ambiguity-Free method – Eq. (8) – and the Linear Ambiguity-Free method – Eq (5).

Data span (mm:ss)	Approx. cords. accuracy ($\varepsilon_{\Delta X}$ $\varepsilon_{\Delta Y}$ $\varepsilon_{\Delta Z}$) (mm)			LAMBDA accuracy ($\varepsilon_{\Delta X}$ $\varepsilon_{\Delta Y}$ $\varepsilon_{\Delta Z}$) (mm)			t_{LAMBDA} (s)	General amb-free accuracy. Eq. (8) ($\varepsilon_{\Delta X}$ $\varepsilon_{\Delta Y}$ $\varepsilon_{\Delta Z}$) (mm)			$t_{amb\text{-}free}$ (s)**	$\hat{\mu}_v$ (mm)	$\hat{\sigma}_v$ (mm)	Linear amb-free accuracy. Eq. (5) ($\varepsilon_{\Delta X}$ $\varepsilon_{\Delta Y}$ $\varepsilon_{\Delta Z}$) (mm)			$t_{Linear\text{-}amb\text{-}free}$ (s)
60:00	-12	8	39	Ref. solution*			0.030	0	0	-1	0.046	-0.2	3.8	-1	0	1	0.013
55:00	-21	8	51	1	0	-1	0.030	1	0	-1	0.040	-0.1	3.7	0	0	0	0.011
50:00	-31	21	53	1	0	-1	0.027	1	0	-1	0.038	-0.1	3.7	0	0	1	0.008
45:00	-54	14	24	1	0	-1	0.025	1	0	-1	0.035	-0.1	3.8	0	0	1	0.007
40:00	-75	38	85	1	0	-1	0.025	2	0	-2	1.780	0.0	3.8	***-30***	***3***	***53***	*0.006*
30:00	-115	57	130	0	0	0	0.019	1	0	-2	1.135	-0.1	3.5	***-101***	***57***	***196***	*0.003*
20:00	-79	9	91	-1	0	-2	0.018	0	0	-3	1.698	-0.2	3.3	***-58***	***7***	***115***	*0.001*
10:00	-105	-109	32	1	-1	-3	0.018	0	-1	-3	1.209	-0.2	3.0	***-67***	***-125***	***74***	*0.0005*
5:00	-226	-98	12	0	-1	-3	0.019	0	-1	-3	0.971	-0.2	2.7	***-210***	***-108***	***45***	*0.0003*
4:00	-274	-133	176	***-361***	***-143***	***337***	*0.018*	1	-2	-4	1.084	-0.2	2.9	***-352***	***-155***	***307***	*0.0003*
3:00	-351	-16	278	***-385***	***-112***	***395***	*0.016*	2	-3	-6	1.018	-0.1	2.5	***-330***	***-37***	***315***	*0.0002*
2:00	-438	-18	535	***-386***	***-111***	***397***	*0.016*	3	-3	-9	1.714	0.2	2.3	***-386***	***-111***	***393***	*0.0001*
1:00	-282	-255	21	***-335***	***-343***	***154***	*0.015*	3	-4	-10	1.885	0.4	2.6	***-318***	***-265***	***135***	*0.0001*
0:30	-178	77	-334	***-200***	***34***	***-401***	*0.013*	3	-2	-9	1.746	0.1	3.1	***-207***	***39***	***-394***	*0.0001*

*Reference solution (in m): ΔX = 20.261; ΔY = -20.033; ΔZ = 9.315

**Time strongly depend on the choice of β




6a
6b
0a
0b
5a
5b
9a
7a
8a
7b=8b=...=nb
4b
2b
10a
11a
2a
4a
3b
3a
1b
1a