GNSS Integer Ambiguity Estimation and Evaluation: LAMBDA and Ps-LAMBDA

Bofeng Li[1], Sandra Verhagen and Peter J.G. Teunissen

Abstract. Successful integer carrier-phase ambiguity resolution is crucial for high precision GNSS applications. It includes both integer estimation and evaluation. For integer estimation, the LAMBDA method has been applied in a wide variety of GNSS applications. The method's popularity stems from its numerical efficiency and statistical optimality. However, before conducting ambiguity resolution, one needs to infer how reliable the fixed solution can expected to be, as incorrect fixed ambiguity solutions often lead to unacceptable positioning errors. In this paper, two Matlab software tools are introduced for the evaluation and integer estimation: Ps-LAMBDA and an updated version of LAMBDA. Evaluation of the integer solution is based on the ambiguity success rate. Since this probability of correct integer estimation is generally difficult to compute, easy-to-use approximations and bounds of the ambiguity success rate are provided by the Ps-LAMBDA software. This success rate tool is valuable not only for inferring whether to fix the ambiguities but also for design and research purposes. For the actual integer estimation, the updated version of the LAMBDA software, provides now more options of integer estimation and integer search, including the search-and-shrink strategy. In addition, the Fixed Failure-rate Ratio Test (FF-RT) and the Fixed Critical-value Ratio Test (FC-RT) are incorporated for users to validate the significance of the fixed solution. Using these two software tools together allows for the combined execution of integer estimation and evaluation, thus benefiting multi-frequency, multi-GNSS applications.

Keywords: LAMBDA · Ps-LAMBDA · Integer Rounding · Integer Bootstrapping · Integer Least-Squares · Search-and-Shrink · Ambiguity Success Rate · Ambiguity Resolution

---

[1]B. Li (✉) · P. Teunissen
GNSS Research Center, Department of Spatial Sciences, Curtin University, Perth, Australia
e-mail: bofeng.li@curtin.edu.au

S. Verhagen
Mathematical Geodesy and Positioning, Delft University of Technology, Delft, The Nertherlands

1. Introduction

All high-precision GNSS (Global Navigation Satellite Systems) applications commonly rely on the very precise GNSS carrier-phase observations with successfully fixed ambiguities [1-5]. Hence ambiguity resolution is the key for precision GNSS applications. It comprises integer ambiguity estimation and evaluation.

A variety of ambiguity resolution methods have been developed since the late 1980s [1-6], of which the LAMBDA (Least squares AMBiguity Decorrelation Adjustment) method has become one of the more popular methods. The method includes a numerically efficient implementation of the Integer Least-Squares (ILS) principle and as such maximizes the probability of successful integer estimation [7]. The key of the LAMBDA method is to find the integer solution based on a decorrelated float ambiguity solution instead of the original one. By means of the decorrelating ambiguity transformation the efficiency of the integer search is significantly improved. For more details on the LAMBDA method, one can refer to [6, 8, 9].

High precision GNSS positioning is only possible if the integer ambiguities are correctly fixed. If the fixed solution is unreliable, it too often leads to unacceptable errors in the positioning result. Therefore, in practice, one should first evaluate (predict) how reliable the fixed solution will be. If the reliability of the integer solution is predicted to be lower than a required threshold, one should not proceed with the ambiguity resolution. To predict the reliability of the integer solution, the success rate is employed, i.e. the probability of correct integer estimation [7, 10, 11]. Unfortunately, exact evaluation of the success rate is generally not feasible. It is therefore necessary to find some good approximations of the success rate. So far, a variety of success rate approximations and bounds have been developed for ILS, integer bootstrapping (IB) and integer rounding (IR) [10-13]. However, up to now, no standard software has been available for conducting such evaluations.

In this contribution, we will introduce two Matlab software tools, Ps-LAMBDA and version 3.0 of LAMBDA. In the Ps-LAMBDA software, lower and upper bounds, as well as Monte-Carlo based approximations, of the ILS, IB and IR ambiguity success rates are provided. This success rate tool is valuable in applications not only for deciding on whether or not to fix the ambiguities but also for design and research purposes. In version 3.0 of LAMBDA, the following new features are incorporated: (i) more ILS search options, including a search-and-shrink; (ii) extension of integer estimation methods, namely IB, IR and partial ambiguity resolution (PAR); and (iii) the Fixed Failure-rate Ratio Test (FF-RT) and Fixed Critical-value Ratio Test (FC-RT) for validating the significance of the integer solution.

2. The four steps of integer ambiguity resolution

Consider the GNSS mixed integer linear model

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{a} + \boldsymbol{B}\boldsymbol{b} + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{a} \in \mathbb{Z}^n$ and $\boldsymbol{b} \in \mathbb{R}^p$ are the integer and real parameter vectors, respectively; Their design matrices are $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{B} \in \mathbb{R}^{m \times p}$ with $[\boldsymbol{A}\ \boldsymbol{B}]$ full column rank. $\boldsymbol{y} \in \mathbb{R}^m$ is the observation vector contaminated by the normally distributed random noise $\boldsymbol{e}$ with zero means and variance matrix $\boldsymbol{Q}_{yy}$. This mixed integer model is usually solved in four steps.

1. **Float solution**: In the first step, the integer nature of the ambiguities is discarded and a standard least-squares (LS) parameter estimation is performed. As a result, one obtains the so-called float solution, together with its variance-covariance matrix

$$\begin{bmatrix} \hat{\boldsymbol{a}} \\ \hat{\boldsymbol{b}} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix}, \begin{bmatrix} \boldsymbol{Q}_{\hat{a}\hat{a}} & \boldsymbol{Q}_{\hat{a}\hat{b}} \\ \boldsymbol{Q}_{\hat{b}\hat{a}} & \boldsymbol{Q}_{\hat{b}\hat{b}} \end{bmatrix} \right) \tag{2}$$

   Other forms than batch least-squares such as recursive LS or Kalman filtering may also be used to come up with a float solution. Such choices will depend on the application and on the structure of the GNSS model.

2. **Integer Solution**: The purpose of this second step is to take the integer constraints $\boldsymbol{a} \in \mathbb{Z}^n$ into account. Hence, a mapping $I : \mathbb{R}^n \mapsto \mathbb{Z}^n$ is introduced that maps the float ambiguities to corresponding integer values

$$\breve{\boldsymbol{a}} = I(\hat{\boldsymbol{a}}) \tag{3}$$

   Many such integer mappings *I* exist. Popular choices are integer rounding (IR), integer bootstrapping (IB) and integer least squares (ILS). ILS is optimal, as it can be shown to have the largest probability of correct integer estimators, i.e. the largest success rate of all integer estimators. IR and IB, however, can also perform quite well, in particular after the LAMBDA decorrelation has been applied. Their advantage over ILS is that no integer search is required.
   The expected quality of $\breve{\boldsymbol{a}} = I(\hat{\boldsymbol{a}})$, as described by the ambiguity success-rate, can be evaluated with the Ps-LAMBDA tool (see next section).

3. **Accept/reject**: Once integer estimates of the ambiguities have been computed, the third step consists of deciding whether or not to accept the integer solution. Several such tests have been proposed and are currently in use in practice. They are all of the form

$$\text{Accept } \breve{\boldsymbol{a}} \quad \text{if} \quad T(\breve{\boldsymbol{a}}) < c \tag{4}$$

   with testing function $T : \mathbb{R}^n \mapsto \mathbb{R}$. The positive scalar c<1 is a tolerance value that needs to be selected by the user. Thus the integer solution $\breve{\boldsymbol{a}}$ is accepted when $T(\breve{\boldsymbol{a}})$ is sufficiently small; otherwise, it is rejected in favour of the float solution $\hat{\boldsymbol{a}}$. Different choices for T can be made. Examples include the ratio-test, the difference-test and the projection-test [14-18]. All these tests can be cast in the framework of Integer Aperture Estimation (IAE) [19-21].The ratio-test is probably one of the most popular. Both the Fixed Critical-value Ratio Test (FC-RT) and the Fixed Failure-rate Ratio Test (FF-RT) are part of the new version of LAMBDA.

4. **Fixed Solution**: In the final step, once $\breve{\boldsymbol{a}}$ is accepted, the float estimator $\hat{\boldsymbol{b}}$ is re-adjusted to obtain the so-called fixed estimator

$$\breve{\boldsymbol{b}} = \hat{\boldsymbol{b}} - \boldsymbol{Q}_{\hat{b}\hat{a}} \boldsymbol{Q}_{\hat{a}\hat{a}}^{-1} \left(\hat{\boldsymbol{a}} - \breve{\boldsymbol{a}} \right) \tag{5}$$

   This solution has a quality that is commensurate with the high precision of the phase data, *provided* that the uncertainty in the random integer vector $\breve{\boldsymbol{a}}$ can be neglected.
   To decide whether or not the uncertainty in $\breve{\boldsymbol{a}}$ can be neglected, an evaluation of its probabilistic properties is essential.

3 Ps-LAMBDA software for ambiguity success rate evaluation

3.1 The ambiguity success rate

With the Ps-LAMBDA tool one can evaluate the expected quality of $\breve{\boldsymbol{a}} = I(\hat{\boldsymbol{a}})$, as described by its success-rate. The success rate of $\breve{\boldsymbol{a}}$ is defined as the integral of the probability density function (PDF) of the float solution over the pull-in region S_a, where $\boldsymbol{a}$ is the correct integer vector:

$$P_{\mathrm{s}} = P\left(\breve{\boldsymbol{a}} = \boldsymbol{a} \right) = P\left(\hat{\boldsymbol{a}} \in S_a \right) = \int_{S_a} f_{\hat{\boldsymbol{a}}}(\boldsymbol{x} \mid \boldsymbol{a}) \, \mathrm{d}\boldsymbol{x} \tag{6}$$

The PDF of the float solution is

$$f_{\hat{\boldsymbol{a}}}(\boldsymbol{x} \mid \boldsymbol{a}) = \frac{1}{\sqrt{\det\left(2\pi \boldsymbol{Q}_{\hat{a}\hat{a}}\right)}} \exp\left\{ -\frac{1}{2} (\boldsymbol{x} - \boldsymbol{a})^T \boldsymbol{Q}_{\hat{a}\hat{a}}^{-1} (\boldsymbol{x} - \boldsymbol{a}) \right\} \tag{7}$$

As the pull-in regions of the integer estimators are integer translation invariant, the success rate can be evaluated, without knowledge of the unknown ambiguity vector, as:

$$P_{\mathrm{s}} = \int_{S_0} f_{\hat{\boldsymbol{a}}}(\boldsymbol{x} \mid 0) \, d\boldsymbol{x} \tag{8}$$

This shows that the success rate depends on the PDF and on the pull-in region. The PDF is captured by variance matrix $\boldsymbol{Q}_{\hat{a}\hat{a}}$, while the pull-in region is specified by the chosen integer estimation method. For more detail on the pull-in regions of ILS, IB and IR methods, one can refer to [7]. For the success rate evaluation, only the variance matrix $\boldsymbol{Q}_{\hat{a}\hat{a}}$ is needed and not the float ambiguity solution $\hat{\boldsymbol{a}}$ itself.

3.2 Success rate approximation and bounds

First of all, one can evaluate the success rate (8) by making use of Monte Carlo simulations. The procedure is as follows. One generates a random sample $\hat{\boldsymbol{a}}$ from the distribution $\mathrm{N}(0, \boldsymbol{Q}_{\hat{a}\hat{a}})$ and uses it as input for integer estimation. If the output of this estimator equals the null vector, then it is correct, otherwise it is incorrect. This process can be repeated an N number of times, and one can count how many times the null vector is obtained as a solution, say N_s times. The approximation of the success rate follows then as:

$$P_s = N_s / N \tag{9}$$

In order to get good approximations, the number of samples N must be sufficiently large. The disadvantage is that it may be very time-consuming, especially in case of ILS, since for each sample an integer search is required. Therefore the Ps-LAMBDA tool also provides easy-to-compute lower and upper bounds of the ambiguity success rate

So far there are a variety of lower and upper bounds in literatures. Some of them are based on simplifying the complete variance matrix of ambiguities and some of them based on simplifying the complicated pull-in region [22]. Besides, the relation of success rate between ILS, IB and IR methods can be used [7]:

$$P\left(\breve{\boldsymbol{a}}_{\mathrm{IR}} = \boldsymbol{a}\right) \le P\left(\breve{\boldsymbol{a}}_{\mathrm{IB}} = \boldsymbol{a}\right) \le P\left(\breve{\boldsymbol{a}}_{\mathrm{ILS}} = \boldsymbol{a}\right) \tag{10}$$

Since this ordering is the same as the ordering in terms of complexity, one may use IB success rate as lower bound of ILS and upper bound of IR or use IR as lower bound of IB. The extensive experience studies indicated that the IR and IB bounds work well based on the decorrelated ambiguities, especially, the IB success rate is a sharp lower bound of ILS. An evaluation for some of the bounds was made in [13, 23].

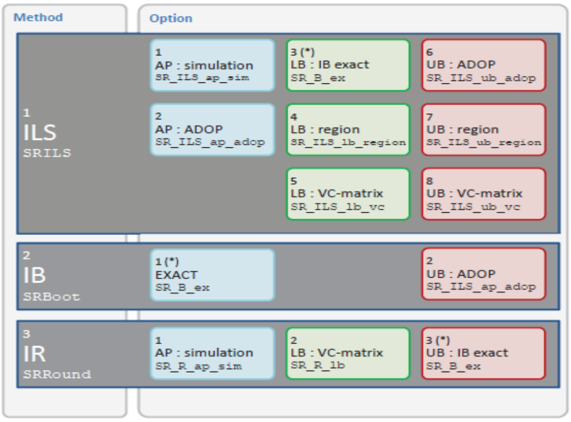
3.3 Ps-LAMBDA software and its demonstration

The Monte Carlo based simulations, as well as all lower and upper bounds of the success rate are now implemented in the Ps-LAMBDA software for ILS, IB and IR methods. Fig. 1 gives an overview of the software's structure. The main routine is **SuccessRate** which needs the inputs:

Qa	The variance matrix of the float solution $\boldsymbol{Q}_{\hat{a}\hat{a}}$
method	1=ILS [DEFAULT], 2=IB, 3=IR
option	The approximation / bound to compute (see Fig. 1)
decor	1=decorrelation [DEFAULT], 0=no decorrelation
nsamp	Number of samples used for simulation-based approximation

The choice for **decor** is only relevant to IR and IB, since these estimators are not Z-invariant. Decorrelation is always applied for ILS to ensure computational efficiency.

Fig. 1 Ps-LAMBDA: overview of available methods and options in routine **SuccessRate**. Default option is indicated with (*). Names of underlying routines are shown as well. AP = approximation, LB = lower bound, UB = upper bound



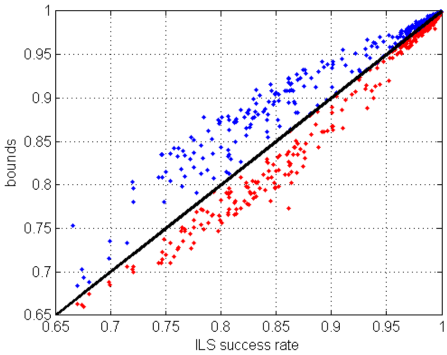


Fig. 2 ILS success rates: upper bound based on region (blue) and lower bound based on IB (red) versus the actual ILS success rate

Fig. 2 shows the performance of the Monte-Carlo simulation based success rate, the IB based lower bound and the region based upper bound. They are corresponding to options 1, 3 and 7 in Fig. 1. This result is for a CORS baseline ambiguity resolution with ionosphere-weighted model. The standard deviation (STD) of single-differenced ionosphere constraint is 7cm. The 24 hour dual-frequency GPS observations are used with STDs of phase and code 20 cm and 2mm, respectively. The number of epochs varies from 1 to 5 with 5s sampling interval to obtain the success rate results from 0.65 to 1. The number of samples used for the simulation based success rate is 10^6. With this number, the approximation will be very close to the true value [25]. The result shows that both lower and upper bounds perform very well, particularly when the success rate is larger than 0.95. Considering that in the real applications, one usually accepts the fixed solution only when its success rate is close to 1 (say, >99%), one can use these two bounds to easily evaluate the success rate. For more information about the performance of the other approximations and bounds, one can refer to [13, 23, 24].

4. LAMBDA (version 3) for ambiguity estimation and validation

In the previous educational version of the LAMBDA software, only the ILS method was implemented and the search was executed by enumerating all integer candidates inside the search ellipsoid with a small, but fixed ellipsoid size. This search strategy can become time-consuming in the high-dimensional case with multi-frequency, multi-GNSS applications. Moreover, one may alternatively prefer simpler estimation method, like for instance IB or IR, in case their success rate is high enough. The main features of the new LAMBDA version 3 software are as follows:

- Additional to the enumeration with fixed ellipsoid size, another search strategy was embedded based on searching in an alternating way around the conditional estimates and shrinking the search ellipsoid. This concept was already presented in [6], see also [8], but not implemented in the previous educational versions of LAMBDA.
- As alternative to ILS, one may now also choose the IB or IR estimators.
- It is possible to output the IB success rate. This success rate is known to be a tight lower bound of the ILS success rate [7].
- Partial Ambiguity Resolution (PAR) can be applied, based on fixing a subset of the decorrelated ambiguities such that the success rate will be larger or equal to a minimum required success rate [25].
- The Ratio Test can be applied to validate the significance of the fixed solution. The model-driven Fixed Failure-rate Ratio Test (FF-RT) or the classical Ratio Test with a fixed (user-defined) threshold can be applied.
- A modular approach with a limited set of subroutines, enabling one to perform for example only the decorrelation-step, only the search-step, or both steps.

We now describe and geometrically illustrate the concept of the search-and-shrink approach. For the other features, one may refer to the listed references and the software manual.

4.1 ILS Search space

Using the full information of $\boldsymbol{Q}_{\hat{a}\hat{a}}$, the ILS ambiguity resolution is defined as

$$\breve{\boldsymbol{a}} = \arg\min \ (\boldsymbol{z}-\hat{\boldsymbol{a}})^T \boldsymbol{Q}_{\hat{a}\hat{a}}^{-1} (\boldsymbol{z}-\hat{\boldsymbol{a}}), \quad \forall \boldsymbol{z} \in \mathbb{Z}^m \tag{11}$$

The integer minimizer (11) is obtained through a search over the integer grid points of an *n*-dimensional hyper-ellipsoid defined by

$$F(\boldsymbol{z}) = (\boldsymbol{z}-\hat{\boldsymbol{a}})^T \boldsymbol{Q}_{\hat{a}\hat{a}}^{-1} (\boldsymbol{z}-\hat{\boldsymbol{a}}) \le \chi^2 \tag{12}$$

The integer grid point $\boldsymbol{z}$ inside the hyper-ellipsoid which gives the minimum value of function $F(\boldsymbol{z})$ is the optimal ILS solution $\breve{\boldsymbol{a}}$. The search efficiency is governed by the size χ^2 and the shape of ellipsoid. The constant χ^2 can be predetermined using different strategies and can also be shrunken during the search [8, 26]. The shape and orientation of the ellipsoid are defined by the variance matrix $\boldsymbol{Q}_{\hat{a}\hat{a}}$. The high correlated variance matrix will often lead to search halting. To improve the search efficiency, the decorrelated variables, $\hat{\boldsymbol{z}} = \boldsymbol{Z}^T \hat{\boldsymbol{a}}$ and $\boldsymbol{Q}_{\hat{z}\hat{z}} = \boldsymbol{Z}^T \boldsymbol{Q}_{\hat{a}\hat{a}} \boldsymbol{Z}$ are used in (12) instead of $\hat{\boldsymbol{a}}$ and $\boldsymbol{Q}_{\hat{a}\hat{a}}$, respectively.

4.2 Search-and-shrink in two dimensions

In order to explain the concept of the search-and-shrink procedure, we describe it with a two-dimensional (2D) example. Figures 3 and 4 show how the search procedures work with enumeration and shrinking for a 2D example. The float solution is depicted with a *blue* asterisk. The grey line shows the line defined as

$$\hat{z}_{1|2} = \hat{z}_1 - \sigma_{\hat{z}_1\hat{z}_2} \sigma_{\hat{z}_2\hat{z}_2}^{-2} \left(\hat{z}_2 - z_2 \right) \tag{13}$$

It shows that if $\hat{z}_2$ is rounded to its nearest integer 0, the conditional estimate $\hat{z}_{1|2}$ will be the intersection of the grey line with the grid line at z_2=0. With bootstrapping, this conditional estimate is then rounded to its nearest integer 1. If only 2 integer vectors are requested, both enumeration and shrinking search strategies will give the same initial search ellipse, see left panel of Fig. 3. If 6 candidate integer vectors are requested, the difference between the two search strategies becomes clear. Figure 3 shows on the right the search ellipse for the enumerating search strategy. All 16 grid points inside ellipse are examined to choose 6 candidates (*black dots*) with the smallest $F(\mathbf{z}^{(i)})$.

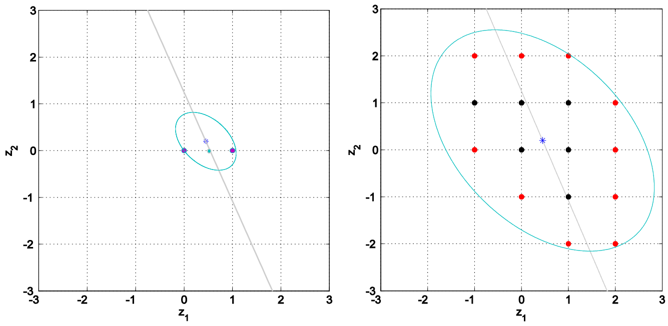


Fig. 3 Search ellipse for 2D example for enumerating search strategy. The requested number of integer vectors is 2 (*left*) or 6 (*right*).

With the search-and-shrink strategy, the 6-step procedure is described and each step is shown as a separate panel in Fig. 4. For each step the new candidate is shown in *blue* in the corresponding panel. Candidates from previous steps are shown in *black*. If at a certain stage a candidate is removed, as it is outside the shrunken ellipse, it is shown in *red*. For every step the old search ellipse (*red*) and the new shrunken ellipse (*blue*) are shown if shrinking is possible.

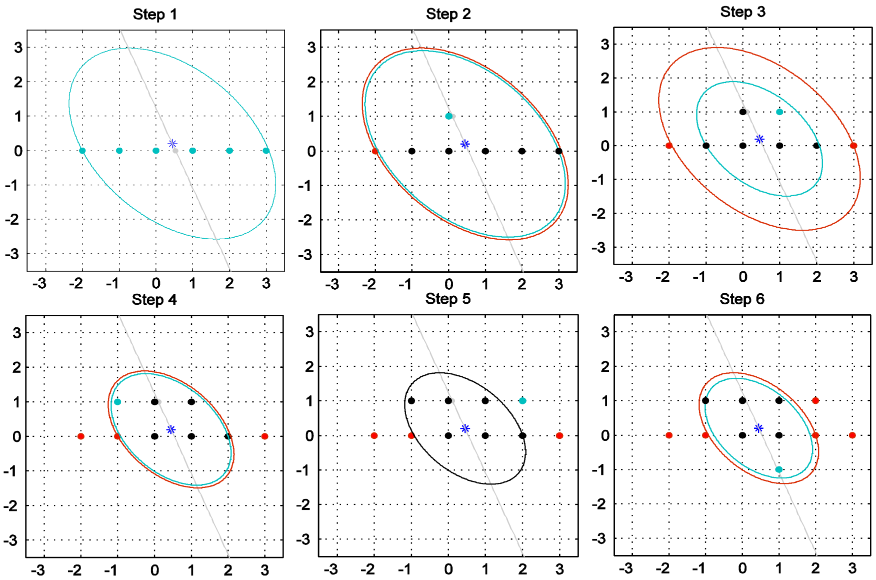


Fig. 4 Search-and-shrink procedure for 2D example (The *x* and *y* axes denote the ambiguity z_1 and z_2, respectively). The requested number of integer vectors is 6.

1. With the bootstrapped solution $[1\ 0]^T$ as the first candidate, we find the next 5 candidates by rounding the conditional ambiguity $\hat{z}_{1|2}$ to its 2nd, 3rd, 4th and 5th nearest integer. We now have 6 candidates $\boldsymbol{z}(i)$ (blue points), and the size of the search ellipse is set to the maximum $F(\boldsymbol{z}(i))$, see the blue ellipse.
2. Round $\hat{z}_2$ to its second nearest integer (z_2=1), and round the corresponding new conditional estimate of the first ambiguity to the nearest integer. This will give you a new candidate (blue point in panel 2). It resides in the search ellipse. Therefore, the integer candidate with the largest $F(\boldsymbol{z}(i))$ is removed (see red point), and the ellipse size is now set to the largest value of the remaining candidates.
3. Next, round the conditional estimate of the first ambiguity from Step 2 to the second nearest integer. Again it resides in the shrunken search ellipsoid. Therefore, the integer candidate with the largest $F(\boldsymbol{z}(i))$ is removed, and the ellipse size is now set to the largest value of the remaining candidates.
4. Round the conditional estimate of the first ambiguity from Step 2 to the third nearest integer. Again it resides in the shrunken search ellipsoid. Therefore, the integer candidate with the largest $F(\boldsymbol{z}(i))$ is removed, and the ellipse size is now set to the largest value of the remaining candidates.

5. Round the conditional estimate of the first ambiguity from Step 2 to the fourth nearest integer. This candidate is outside the search ellipsoid and is disregarded. No shrinking in this step.
6. A new candidate for the second ambiguity is obtained by rounding $\hat{z}_2$ to the third nearest integer (z_2=−1). Rounding the corresponding new conditional estimate of the first ambiguity to the nearest integer results in a new candidate, which resides in the search ellipse. Therefore, the integer candidate with the largest $F(\mathbf{z}(i))$ is removed, and the ellipse size is now set to the largest value of the remaining candidates.

For the *n*-dimensional case the search-and-shrink strategy goes along similar lines.

5 Summary

In this paper we introduced two Matlab software tools, Ps-LAMBDA and LAMBDA (version 3). They are developed for integer ambiguity evaluation and estimation, respectively. With Ps-LAMBDA the ambiguity success rates, and their bounds, of ILS, IB and IR can be computed, while the new version of LAMBDA provides more options for integer estimation, including the fixed failure-rate ratio test. The new LAMBDA (version 3) software tool can be downloaded from http://saegnss2.curtin.edu.au/~gnssweb/index.php?request=getlambda

Acknowledgments This work was done in the framework of the project 'New Carrier-Phase Processing Strategies for Next Generation GNSS Positioning' of the Cooperative Research Centre for Spatial Information (CRC-SI2). Professor Teunissen is the recipient of an Australian Research Council Federation Fellowship (project No. FF0883188). All this support is gratefully acknowledged.

References

1. Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J. (2001). *Global positioning system: theory and practice*. 5th edn. Springer Berlin Heidelberg, New York.
2. Leick, A. (2004). *GPS satellite surveying*. 3rd edn. John Wiley, New York.
3. Teunissen, P. J. G., Kleusberg, A. (1998). *GPS for geodesy*. 2nd edn. Springer Berlin Heidelberg New York.
4. Blewitt, G. (1989). Carrier phase ambiguity resolution for the global positioning system applied to geodetic baselines up to 2000 km. *Journal of Geophysical Research*, 94, 135–151.
5. Dong, D., Bock, Y. (1989). Global positioning system network analysis with phase ambiguity resolution applied to crustal deformation studies in California. *Journal of Geophysical Reseach* 94 (B4), 3949–3966.
6. Teunissen, P.J.G. (1993). Least squares estimation of the integer GPS ambiguities. In: Invited lecture, Section IV Theory and Methodology, IAG General Meeting, Beijing.
7. Teunissen, P. J. G., (1999). An optimality property of the integer least-squares estimator. *Journal of Geodesy*, 73 (11), 587–593.

8. De Jonge, P. J., Tiberius, C.C.J.M. (1996). The LAMBDA method for integer ambiguity estimation: implementation aspects, LGR-Series, No 12. Tech. report, Delft University of Technology.
9. Li, B., Teunissen, P.J.G. (2011). High dimensional integer ambiguity resolution: A first comparison between LAMBDA and Bernese. *The Journal of Navigation*, 64, S192–S210
10. Hassibi, A., Boyd, S. (1998). Integer parameter estimation in linear models with applications to GPS. *IEEE Transactions on Signal Processing*, 46(11), 2938–2952.
11. Teunissen, P.J.G. (1998). Success probability of integer GPS ambiguity rounding and bootstrapping. *Journal of Geodesy*, 72, 606–612.
12. Teunissen, P.J.G. (2000). ADOP based upper bounds for the bootstrapped and the least-squares ambiguity success rates. *Artificial Satellites*, 35(4), 171–179.
13. Verhagen, S. (2005). On the reliability of integer ambiguity resolution. *Navigation*, 52(2), 99–110.
14. Euler, H.J., Schaffrin, B. (1991). On a measure for the discernibility between different ambiguity solutions in the static-kinematic GPS-mode. Proc of Kinematic Systems in Geodesy, Surveying, and Remote Sensing, International Association of Geodesy Series, Vol.107, pp. 285–295.
15. Han, S. (1997). Quality control issues relating to instantaneous ambiguity resolution for real-time GPS kinematic positioning. *Journal of Geodesy*, 71(6), 351–361.
16. Landau, H., Euler, H.J. (1992). On-the-fly ambiguity resolution for precise differential positioning. Proc. of ION GPS-1992, Albuquerque NM, 607–613.
17. Tiberius, C.C.J.M., De Jonge, P. (1995). Fast positioning using the LAMBDA method. In Proc. of DSNS’95, Bergen, Norway, paper no.30. The Nordic Institute of Navigation, Oslo.
18. Wang, J., Stewart, M.P., Tsakiri, M. (1998). A discrimination test procedure for ambiguity resolution on-the-fly. *Journal of Geodesy*, 72(11), 644–653.
19. Teunissen, P.J.G., Verhagen, S. (2009). The GNSS ambiguity ratio-test revisited: a better way of using it. *Survey Review*, 41(312), 138–151.
20. Verhagen, S., Teunissen, P.J.G. (2006). New global navigation satellite system ambiguity resolution method compared to existing approaches. *Journal of Guidance, Control, and Dynamics*, 29(4), 981–991.
21. Verhagen, S., Teunissen, P.J.G. (2012). The ratio test for future GNSS ambiguity resolution. *GPS Solutions*, doi: 10.1007/s10291-012-0299-z
22. Teunissen, P. J. G., 1998. On the integer normal distribution of the GPS ambiguities. *Artificial Satellites*, 33 (2), 49–64.
23. Thomsen, H. F., 2000. Evaluation of upper and lower bounds on the success probability. Proc. of ION GPS-2000, Salt Lake City UT, 183–188.
24. Verhagen, S., Li, B., and Teunissen, P.J.G., 2012. Ps-LAMBDA: ambiguity success rate evaluation software for interferometric applications, *Computer and Geosciences*, (in press)
25. Verhagen, S., Teunissen, P.J.G., van der Marel, H., Li, B. (2011). GNSS ambiguity resolution: which subset to fix? IGNSS Symposium 2011, Sydney, NSW, Australia.
26. Chang, X., Yang, X., Zhou, T. (2005). MLAMBDA: a modified LAMBDA method for integer least-squares estimation. *Journal of Geodesy*, 79, 552–565.