Journal of Geodesy (2001) 75: 399–407

Journal of
Geodesy

Integer estimation in the presence of biases

P. J. G. Teunissen

Department of Mathematical Geodesy and Positioning, Delft University of Technology,
Thijsseweg 11, 2629 JA Delft, The Netherlands
e-mail: p.j.g.teunissen@geo.tudelft.nl; Tel.: +31-15-278-2558; Fax: +31-15-278-3711

Received: 28 September 2000 / Accepted: 29 March 2001

Abstract. Carrier phase ambiguity resolution is the key to fast and high-precision GNSS (Global Navigation Satellite System) kinematic positioning. Critical in the application of ambiguity resolution is the quality of the computed integer ambiguities. Unsuccessful ambiguity resolution, when passed unnoticed, will too often lead to unacceptable errors in the positioning results. Very high success rates are therefore required for ambiguity resolution to be reliable. Biases which are unaccounted for will lower the success rate and thus increase the chance of unsuccessful ambiguity resolution. The performance of integer ambiguity estimation in the presence of such biases is studied. Particular attention is given to integer rounding, integer bootstrapping and integer least squares. Lower and upper bounds, as well as an exact and easy-to-compute formula for the bias-affected success rate, are presented. These results will enable the evaluation of the bias robustness of ambiguity resolution.

Keywords: GNSS – Ambiguity Resolution – Success Rate – Bias Robustness

1 Introduction

Ambiguity resolution applies to a great variety of GNSS models currently in use. They range from single-baseline models used for kinematic positioning to multi-baseline models used as a tool for studying geodynamic phenomena. An overview of these models, together with their applications in surveying, navigation and geodesy, can be found in textbooks such as those of Hofmann-Wellenhof et al. (1997), Leick (1995), Parkinson and Spilker (1996), Strang and Borre (1997) and Teunissen and Kleusberg (1998).

Despite the differences in application of the various GNSS models, their ambiguity resolution problems are intrinsically the same. Hence, any rigorous method of ambiguity resolution should be applicable to each of these models. Any such method should be able to efficiently obtain integer ambiguity estimates from the 'float' solution, as well as provide the user or analyst with tools to evaluate the quality of the integer solution so obtained. Unfortunately the availability of proper indicators for the qualitative aspects of the integer ambiguity estimators is still lacking in most of the present-day GNSS positioning systems.

It is of importance to be able to evaluate the quality of the integer solution, since unsuccessful ambiguity resolution, when passed unnoticed, will all too often lead to unacceptable errors in the positioning results. We therefore need to have a way of knowing how often we can expect the computed ambiguity solution to coincide with the correct, but unknown, solution. Is this nine out of 10 times, 99 out of 100, or a higher percentage? It will certainly never equal 100%. After all, the integer ambiguities are computed from the data. They are therefore subject to uncertainty just like the data are.

In order to describe the quality of the integer ambiguity solution, we require the probability distribution of the integer estimator. This distribution will be a probability mass function, due to the integer nature of the ambiguities. Of this probability mass function, the probability of correct integer estimation – also referred to as the success rate – is of particular interest. This probability depends on three contributing factors: the functional model, the stochastic model and the chosen method of integer estimation. Changes in any of these will affect the ambiguity success rate.

In this contribution we address the probabilistic aspects of integer ambiguity estimation in the presence of biases. We will refrain however, from discussing the computational intricacies of integer estimation. For a discussion of these aspects, we refer to, for example, Teunissen (1993) and de Jonge and Tiberius (1996a), or to the textbooks of Hofmann-Wellenhof et al. (1997), Strang and Borre (1997) and Teunissen and Kleusberg (1998). A very efficient way of solving the integer estimation problem is provided by the LAMBDA method. A description of the LAMBDA method can be found in the aforementioned publications, while practical results obtained with it can be found in, for example, Boon and Ambrosius (1997), Boon et al. (1997), Cox and Brading (1999), de Jonge and Tiberius (1996b), de Jonge et al. (1996), Han (1995), Jonkman (1998), Peng et al. (1999), Tiberius and de Jonge (1995) and Tiberius et al. (1997).

Since unaccounted biases in the functional model will produce biases in the ambiguity 'float' solution, they will also affect the performance of ambiguity resolution. In order to evaluate the bias robustness of ambiguity resolution, we need to be able to assess the impact these biases have on the probability of correct integer estimation. In this contribution, we present diagnostic tools for evaluating this impact.

This contribution is organized as follows. In Sect. 2 we first present a brief review of the general problem of integer ambiguity estimation. The class of admissible integer estimators is described and conditions for unbiased integer estimation are given. Each member from this class is uniquely defined by its ambiguity pull-in region. It is also shown how these pull-in regions govern the success rates of the admissible integer estimators. Well-known examples of integer estimators that belong to this class are 'integer rounding', 'integer bootstrapping' and 'integer least-squares (LS)'.

In Sect. 3 we study the impact of unaccounted biases. It is shown that biases in the 'float' solution reduce the ambiguity success rate and that this reduction increases with the size of the bias along a fixed direction. We also present different methods for evaluating the bias-affected success rates. For bootstrapping we present an exact and easy-to-evaluate formula for the bias-affected success rate. For other integer estimators, such as 'integer rounding' and 'integer LS', it is generally very difficult to obtain such analytical expressions. For these two estimators, easy-to-compute lower and upper bounds of their bias-affected success rates are given. Finally, it is shown how the method of simulation can be used to compute the bias-affected success rates of arbitrary integer estimators.

In this contribution we will make frequent use of the following notation. The canonical unit vector having as its ith entry a 1, and zero entries otherwise, is denoted as c_i. The weighted norm of a vector x will be denoted as $\| x \|_M$. Thus $\| x \|_M^2 = x^T M^{-1} x$. Also the triangular factorization $Q_{\hat{a}} = LDL^T$ of the ambiguity variance–covariance matrix $Q_{\hat{a}}$ will often be used. Matrices L and D are then, respectively, a unit lower triangular matrix and a diagonal matrix. The diagonal entries of matrix D are the sequential conditional variances of the ambiguities. Thus $D = \text{diag}(\ldots, \sigma^2_{\hat{a}_{i|I}}, \ldots)$, with $\sigma^2_{\hat{a}_{i|I}}$ the variance of the ith LS ambiguity obtained through a conditioning on the previous $I = \{1, \ldots, (i-1)\}$ ambiguities.

2 Integer estimation

2.1 Admissible integer estimators

GNSS ambiguity resolution is the process of resolving the unknown cycle ambiguities of the carrier phase data as integers. When solving for the carrier phase ambiguities, we still have to decide which integer estimation principle to choose. Various choices are possible. In this section we will define the class of admissible integer estimators. Members from this class are uniquely determined by their so-called pull-in regions.

Let $\hat{a} \in R^n$ be the real-valued 'float' solution of the vector of carrier phase ambiguities. Deciding on an integer estimator implies that a mapping $F : R^n \rightarrow Z^n$ is chosen, from the n-dimensional space of reals to the n-dimensional space of integers, such that the corresponding integer vector of ambiguities is obtained as $\check{a} = F(\hat{a})$. Due to the discrete nature of Z^n, the map F will not be one-to-one, but instead a many-to-one map. Hence, different real-valued ambiguity vectors will be mapped to the same integer vector. We can therefore assign a subset $S_z \subset R^n$ to each integer vector z

$$S_z = \{x \in R^n \mid z = F(x)\}, \ z \in Z^n \tag{1}$$

This subset, referred to as the *pull-in region* of z, contains all real-valued ambiguity vectors that will be mapped by F to the same integer vector z. It is the region in which all ambiguity 'float' solutions are pulled to the same 'fixed' ambiguity vector z. Using the pull-in regions, the integer estimator can be defined as $\check{a} = z \Longleftrightarrow \hat{a} \in S_z$. Using the indicator function of the pull-in regions, the integer estimator can then be written as

$$\check{a} = \sum_{z \in Z^n} z s_z(\hat{a}) \text{ with } s_z(\hat{a}) = \begin{cases} 1 & \text{if } \hat{a} \in S_z \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Since the pull-in regions define the integer estimator completely, we can define classes of integer estimators by imposing various conditions on the pull-in regions. In Teunissen (1999), the class of admissible integer estimators was defined as follows.

Definition. The *integer estimator* $\check{a} = \sum_{z \in Z^n} z s_z(\hat{a})$ is said to be *admissible* if

(i) $\cup_{z \in Z^n} S_z = R^n$

(ii) $\text{int} S_{z_1} \cap \text{int} S_{z_2} = \emptyset, \ \forall z_1, z_2 \in Z^n, z_1 \neq z_2$

(iii) $S_z = z + S_0, \ \forall z \in Z^n$

This definition was motivated as follows. Each one of the above three conditions describes a property which it seems reasonable is possessed by an arbitrary integer ambiguity estimator. The first condition states that the pull-in regions should not leave any gaps and the second that they should not overlap. The absence of gaps is required in order to be able to map any 'float' solution $\hat{a} \in R^n$ to Z^n, while the absence of overlaps is required to guarantee that the 'float' solution is mapped to just one integer vector. Note that we allow the pull-in regions to have common boundaries. Only their interiors (int) are not allowed to overlap. The common boundaries are permitted if we assume to have zero probability that $\hat{a}$ lies on one of these boundaries. This will be the case when the probability density function (PDF) of $\hat{a}$ is continuous.

The third and last condition follows from the requirement that $F(x + z) = F(x) + z, \ \forall x \in R^n, \ z \in Z^n$.

This condition is also a reasonable one to ask for. It states that when the 'float' solution is perturbed by an integer vector z, the corresponding integer solution is perturbed by the same amount. This property allows us to apply the *integer remove–restore* technique: $F(\hat{a}-z)+z=F(\hat{a})$. It therefore allows us to work with the fractional parts of the entries of $\hat{a}$, instead of with its complete entries.

2.2 Unbiased integer estimation

Each integer estimator has its own probability distribution. Since the integer estimator is of the discrete type, its distribution will be a probability *mass* function (PMF). It will be denoted as $P(\check{a}=z)$, with $z \in Z^n$. In order to determine this distribution we first need the PDF of $\hat{a}$. It will be denoted as $p_{\hat{a}}(x)$, with $x \in R^n$. Since $\check{a}=z \Longleftrightarrow \hat{a} \in S_z$, it follows that $P(\check{a}=z)=P(\hat{a} \in S_z)$. Hence, the PMF of $\check{a}$ reads

$$P(\check{a}=z)=\int_{S_z} p_{\hat{a}}(x)\mathrm{d}x, \ \forall z \in Z^n \tag{3}$$

The probability masses of the PMF therefore follow from integrating the PDF over the pull-in regions. The PMF of $\check{a}$ can be used to study various properties of the integer estimator. One such property is the possible unbiasedness of the integer estimator. This property will depend on the PDF $p_{\hat{a}}(x)$ and on the geometry of the pull-in regions. In the following we will assume that both the PDF and the pull-in regions are integer-symmetric. The pull-in region S_u is said to be *u-symmetric* if it is symmetric with respect to $u \in Z^n$. That is, $x \in S_u \Longleftrightarrow 2u-x \in S_u$. Similarly, the PDF is said to be u-symmetric if $p_{\hat{a}}(x)=p_{\hat{a}}(2u-x)$, $\forall x \in R^n$. Based on these two properties, we have the following result.

Corollary 1. *If the PDF $p_{\hat{a}}(x)$ and the pull-in region S_u are both u-symmetric, then the mean of the corresponding admissible integer estimator $\check{a}$ equals $u \in Z^n$*

$$E\{\check{a}\}=u \tag{4}$$

Proof. We clearly have $E\{\check{a}\}=u$, when the PMF is symmetric about u: $P(\check{a}=u+z)=P(\check{a}=u-z)$, $\forall z \in Z^n$. The symmetry of the PMF is shown as follows:

$$\begin{aligned}P(\check{a}=u+z) &= \int_{S_{u+z}} p_{\hat{a}}(x)\mathrm{d}x = \int_{S_z} p_{\hat{a}}(y+u)\mathrm{d}y \\ &= \int_{S_z} p_{\hat{a}}(u-y)\mathrm{d}y = \int_{S_{-z}} p_{\hat{a}}(u+y)\mathrm{d}y \\ &= \int_{S_{u-z}} p_{\hat{a}}(x)\mathrm{d}x = P(\check{a}=u-z)\end{aligned}$$

The second and fifth equations follow from the translational property of pull-in regions, while the third and the fourth equations follow from, respectively, the symmetry of the PDF and the symmetry of the pull-in regions. □

The above result can now be applied to the integer estimator of the carrier phase ambiguities. It is usually assumed that the 'float' solution is unbiased and normally (Gaussian) distributed: $\hat{a} \sim N(a, Q_{\hat{a}})$, with $a \in Z^n$ the true, but unknown, integer ambiguity vector, and $Q_{\hat{a}}$ the variance–covariance matrix of the 'float' solution. The PDF of the 'float' solution is then symmetric with respect $a \in Z^n$. From the above corollary it then follows that any integer ambiguity estimator, for which the pull-in regions are integer-symmetric, is unbiased.

Although it is nice to know whether an integer estimator $\check{a}$ is unbiased or not, the property of unbiasedness does not reveal too much about the shape of the probability mass function. It only implies that the total probability mass is evenly (not necessarily symmetrically) distributed about $a \in Z^n$. The property of 'unbiasedness' therefore implies only that we will obtain the correct solution 'on the average'. For GNSS ambiguity resolution, however, this is not enough. In order to be able to assess the reliability of ambiguity resolution, we also need to know how often we can expect the integer solution to coincide with the correct integer. For that we need to know the probability of correct integer estimation. This probability, also referred to as the *ambiguity success rate*, is given as

$$P(\check{a}=a)=\int_{S_a} p_{\hat{a}}(x)\mathrm{d}x \tag{5}$$

The success rate is a very important diagnostic measure for assessing the reliability of ambiguity resolution. It is particularly important because of the way estimated integer ambiguities are treated in practice. In the processing steps following integer estimation, it is common practice to consider the integer ambiguity solution to be deterministic. However, this is only permitted when the success rate is sufficiently large, i.e. when it is sufficiently close to 1. Only then will we be allowed to assume safely that the uncertainty in the integer ambiguity solution can be neglected. A too low success rate will too often lead to incorrect estimates of the integer vector and consequently to unacceptable errors in the GNSS positioning results.

Note that the ambiguity success rate exists whether the integer estimator is biased or not. In the next section we will evaluate the success rate in the presence of biases.

3 The bias effect

3.1 The success rate in the presence of biases

In order to discuss the impact biases have on the performance of ambiguity resolution, we start by assuming that the 'float' solution is biased. We therefore assume that the 'float' solution is distributed as

$$\hat{a} \sim \mathrm{N}(a+b, Q_{\hat{a}}), \ \text{with } a \in Z^n, \ b \in R^n \tag{6}$$

The vector a equals the true but unknown integer ambiguity vector, and b denotes the real-valued bias vector. Biases in the 'float' solution will generally occur when the assumptions underlying its computation are mis-specified. Mis-specifications in the functional model, for instance, will generally lead to a biased 'float' solution. In the case of GNSS, such biases could be generated by outliers in the code data, cycle slips in the phase data, multipath or the presence of unaccounted atmospheric delays.

With Eq. (6), the corresponding ambiguity success rate is given as

$$P_b(\check{a}=a) = \int_{S_a} (2\pi)^{-\frac{n}{2}} \sqrt{\det(Q_{\hat{a}}^{-1})} \times \exp\left\{-\frac{1}{2} \parallel x-a-b \parallel^2_{Q_{\hat{a}}}\right\} \mathrm{d}x \qquad (7)$$

The success rate has been given the subscript b to show the success rate's dependence on the bias. Note that the success rate, when considered as function of the bias vector b, is symmetric about the origin. Also note that the success rate is independent of the unknown integer ambiguity vector a. This is due to the translational property of the pull-in regions and of the PDF of $\hat{a}$. Thus instead of integrating the PDF over the pull-in region S_a, we might as well centre the PDF at b and integrate over S_0.

The following corollary shows the effect biases have on the ambiguity success rate.

Corollary 2. *Let the 'float' solution $\hat{a}$ be distributed as in Eq.* (6) *and let the integer estimator $\check{a}$ be admissible. If the pull-in regions of $\check{a}$ are integer-symmetric and convex, then*

$$P_{b=0}(\check{a}=a) > P_{b\neq 0}(\check{a}=a) > P_{\mu b\neq 0}(\check{a}=a), \ \forall \mu > 1 \qquad (8)$$

Proof. The proof follows from applying a theorem due to Anderson (1955). Anderson's theorem states that, if $f(x): R^n \rightarrow [0,\infty)$ is symmetric about the origin and unimodal, and $A \subset R^n$ is symmetric about the origin and convex, then

$$\int_{A+\lambda y} f(x)\mathrm{d}x > \int_{A+y} f(x)\mathrm{d}x$$

for any $y \in R^n$ and any $\lambda \in [0,1)$. In order to obtain Eq. (8), we set $A = S_0$, $\lambda y = b$, $y = \mu b$ and $f(x) \propto \exp\{-\frac{1}{2} \parallel x \parallel^2_{Q_{\hat{a}}}\}$, and easily verify that $f(x)$ is both symmetric about the origin and unimodal. □

This result confirms that the performance of ambiguity resolution is degraded by the presence of unaccounted biases. Biases in the 'float' solution reduce the ambiguity success rate and this reduction increases with the size of the bias along a fixed direction. The above result does not state, however, by how much the success rate is reduced due to the presence of the bias. This reduction could be unacceptably large or it could be so small that it is negligible.

That the success rate can still be large enough, even in the presence of biases, can be seen as follows. The success rate equals the integral of the PDF of $\hat{a}$ over the pull-in region. In the absence of any bias, the PDF of $\hat{a}$ will be centred at $a \in Z^n$. However, when bias is present it will translate over b and be centred at $a+b$. The translation over b will reduce the success rate. However, if the PDF is sufficiently peaked, the success rate could still be large enough, even though the PDF is now centred at $a+b$. In fact, if the PDF is sufficiently peaked, the success rate will not change by much, provided the vector $a+b$ remains located within the pull-in region. A dramatic drop in the success rate's value will then only occur when the vector $a+b$ crosses the boundary of the pull-in region.

The fact that the success rate needs to be close to one, combined with the fact that the success rate gets reduced due to the presence of unaccounted biases, again emphasizes the importance of being able to evaluate the ambiguity success rates. In the next and following sections we will show how the success rates can be computed in the presence of biases. These results then allow us to study and evaluate the bias robustness of ambiguity resolution. We first consider the integer ambiguity estimator obtained by bootstrapping.

3.2 The bias-affected bootstrapped success rate

Integer bootstrapping is an often-used method of integer ambiguity estimation. It is a simple method, which, when combined with the decorrelation process of the LAMBDA method, can achieve good results. The method is a generalization of the 'integer rounding' method and it goes as follows. If n ambiguities are available, we start with the first ambiguity $\hat{a}_1$ and round its value to the nearest integer. Having obtained the integer value of this first ambiguity, the real-valued estimates of all remaining ambiguities are then corrected by virtue of their correlation with the first ambiguity. Then the second, but now corrected, real-valued ambiguity estimate is rounded to its nearest integer. Having obtained the integer value of the second ambiguity, the real-valued estimates of all remaining $n-2$ ambiguities are then again corrected, but now by virtue of their correlation with the second ambiguity. This process is continued until all ambiguities are taken care of. In essence this 'bootstrapping' technique boils down to the use of a sequential conditional LS adjustment, with a conditioning on the integer ambiguity values obtained in the previous steps. Hence, the bootstrapped estimator is given as

$$\check{a}_{\mathrm{B}} = ([\hat{a}_1], [\hat{a}_{2|1}], \ldots, [\hat{a}_{n|N}])^T \qquad (9)$$

in which $[\cdot]$ denotes the operation of rounding to the nearest integer and $\hat{a}_{i|I}$ is the ith real-valued LS ambiguity obtained through a conditioning on the previous $I = \{1, \ldots, (i-1)\}$ ambiguities. As it was shown in Teunissen (1999), the bootstrapped estimator is an admissible integer estimator of which the origin-centred pull-in region is given as

$$S_{\mathrm{B},0} = \{x \in R^n \mid \mid c_i^T L^{-1} x \mid \leq \tfrac{1}{2}, \ i = 1, \ldots, n\} \qquad (10)$$

The bootstrapped pull-in regions are integer-symmetric, convex subsets, which have a volume equal to 1. In two dimensions, they reduce to parallellograms. From Corollary 1 it follows that the bootstrapped estimator is unbiased in the case that the 'float' solution is unbiased and normally distributed. In the presence of biases, the bootstrapped success rate is given as

$$P_b(\check{a}_{\mathrm{B}} = a) = \int\limits_{S_{\mathrm{B},0}} (2\pi)^{-\frac{n}{2}} \sqrt{\det Q_{\hat{a}}^{-1}} \times \exp\left\{-\frac{1}{2}(x-b)^T Q_{\hat{a}}^{-1}(x-b)\right\} \mathrm{d}x \qquad (11)$$

Since the conditions of Corollary 2 apply, the bootstrapped success rate gets smaller when unaccounted biases are present in the 'float' solution. The following theorem gives an exact and easy-to-compute expression for the bias-affected bootstrapped success rate.

Theorem 1 (*the bias-affected bootstrapped success rate*). *Let $\hat{a}$ be distributed as $N(a+b, Q_{\hat{a}})$, $a \in Z^n$, $b \in R^n$, and let $\check{a}_{\mathrm{B}}$ be the corresponding integer bootstrapped estimator. Then*

$$P_b(\check{a}_{\mathrm{B}} = a) = \prod_{i=1}^{n} \left[\Phi\left(\frac{1-2\beta_i}{2\sigma_{\hat{a}_{i|I}}}\right) + \Phi\left(\frac{1+2\beta_i}{2\sigma_{\hat{a}_{i|I}}}\right) - 1\right] \qquad (12)$$

with β_i the ith entry of the bias vector $L^{-1}b$, $\sigma^2_{\hat{a}_{i|I}}$ the variance of the ith LS ambiguity obtained through a conditioning on the previous $I = \{1, \ldots, (i-1)\}$ ambiguities, and

$$\Phi(x) = \int\limits_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}v^2\right\} \mathrm{d}v$$

The unit lower triangular matrix L follows from the factorization $Q_{\hat{a}} = LDL^T$, with the diagonal matrix $D = \mathrm{diag}(\ldots, \sigma^2_{\hat{a}_{i|I}}, \ldots)$.

Proof. We will first transform the integral of the bootstrapped success rate into a simpler form. As the transformation we choose $f : x = Ly$, with L the triangular factor of $Q_{\hat{a}} = LDL^T$. Then

$$P_b(\check{a}_{\mathrm{B}} = a) = \int\limits_{f^{-1}(S_{\mathrm{B},0})} (2\pi)^{-\frac{n}{2}} \sqrt{\det D^{-1}} \times \exp\left\{-\frac{1}{2}(y - L^{-1}b)^T D^{-1}(y - L^{-1}b)\right\} \mathrm{d}y$$

with the transformed pull-in region

$$f^{-1}(S_{\mathrm{B},0}) = \{y \in R^n \mid | c_i^T y | \leq \tfrac{1}{2}, i = 1, \ldots, n\}$$

Recognizing that D is a diagonal matrix and that the transformed pull-in region has become an origin-centred cube with all sides equal to 1, we may write the above multivariate integral as a product of one-dimensional integrals

$$P_b(\check{a}_{\mathrm{B}} = a) = \prod_{i=1}^{n} \int\limits_{|y_i| \leq \frac{1}{2}} \frac{1}{\sigma_{\hat{a}_{i|I}}\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{y_i - \beta_i}{\sigma_{\hat{a}_{i|I}}}\right)^2\right\} \mathrm{d}y_i = \prod_{i=1}^{n} \int\limits_{-\frac{1+2\beta_i}{2\sigma_{\hat{a}_{i|I}}}}^{\frac{1-2\beta_i}{2\sigma_{\hat{a}_{i|I}}}} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}v^2\right\} \mathrm{d}v$$

from which the result follows.

The above result can now be used to evaluate the bias robustness of carrier phase ambiguity resolution. In order to evaluate $P_b(\check{a}_{\mathrm{B}} = a)$ we need the bias vector b together with the triangular factorization $Q_{\hat{a}} = LDL^T$. The entries of the diagonal matrix D provide the conditional variances, and the triangular factor L, together with b, determines the bias components β_i.

The outcome of bootstrapping is known to depend on the chosen ambiguity parametrization. Bootstrapping of double-differenced (DD) ambiguities, for instance, will produce an integer solution which generally differs from the integer solution obtained from bootstrapping of reparametrized ambiguities. Since this dependency also holds true for the bootstrapped success rate, we should ensure, when using bootstrapping, that an appropriate parametrization of the ambiguities is used. The method of bootstrapping is known to perform poorly, for instance, when applied to the DD ambiguities. This is due to the usually high correlation between the DD ambgiuities. The method should therefore only be applied in combination with the decorrelating Z-transformation of the LAMBDA method (Teunissen 1993; de Jonge and Tiberius 1996a). When this transformation is applied, we work with the decorrelated ambiguity vector $\hat{z} = Z^T\hat{a}$, instead of with the original ambiguity vector $\hat{a}$. But when the original ambiguities are biased, the transformed ambiguities will be biased too. Instead of Eq. (6), we will have

$$\hat{z} \sim N(z + Z^T b, Z^T Q_{\hat{a}} Z) \qquad (13)$$

with z the true, but unknown, transformed integer ambiguity vector and $Z^T b$ the transformed bias vector. The above theorem can then be applied again to obtain the bias-affected success rate $P_b(\check{z}_{\mathrm{B}} = z)$.

For integer estimators other than the bootstrapped estimator, it is usually very difficult to obtain exact and easy-to-evaluate expressions for the success rate. The reason for this lies in the combination of the pull-in region's geometry and the non-zero correlation of the ambiguities. For such estimators we will therefore have to consider alternative means of evaluating the ambiguity success rates. One such approach is discussed in the next section.

3.3 Lower and upper bounds

In this section we will develop easy-to-compute lower and upper bounds for the bias-affected ambiguity success rate. Since the success rate

$$P_b(\check{a} = a) = \int\limits_{S_0} (2\pi)^{-\frac{n}{2}} \sqrt{\det Q_{\hat{a}}^{-1}} \exp\left\{ -\frac{1}{2} \| x - b \|_{Q_{\hat{a}}}^2 \right\} \mathrm{d}x \tag{14}$$

is equal to the probability that a random vector $x \in R^n$, with distribution $x \sim N(b, Q_{\hat{a}})$, lies in the origin-centred pull-in region S_0, we have $P_b(\check{a} = a) = P(x \in S_0)$. To determine lower bounds and upper bounds of the success rate, the idea is to construct subsets L_0 and U_0 such that $L_0 \subset S_0 \subset U_0$. Then

$$P(x \in L_0) \le P_b(\check{a} = a) \le P(x \in U_0) \tag{15}$$

Both types of subsets, L_0 and U_0, should of course be chosen such that the corresponding probabilities are easily evaluated. We will determine such subsets for the following three integer estimators: integer rounding ($\check{a}_{\mathrm{R}} = ([\hat{a}_1], [\hat{a}_2], \ldots, [\hat{a}_n])^T$), integer bootstrapping ($\check{a}_{\mathrm{B}} = ([\hat{a}_1], [\hat{a}_{2|1}], \ldots, [\hat{a}_{n|N}])^T$) and integer LS ($\check{a}_{\mathrm{LS}} = \arg\min_{z \in Z^n} \| \hat{a} - z \|_{Q_{\hat{a}}}^2$). We have included the bootstrapped estimator, because of its close relation to integer rounding. The pull-in regions of these three estimators are given as [see Teunissen 1999]

$$\begin{aligned} S_{\mathrm{R},z} &= \{x \in R^n \mid | c_i^T (x - z) | \le \tfrac{1}{2}, i = 1, \ldots, n\} \\ S_{\mathrm{B},z} &= \{x \in R^n \mid | c_i^T L^{-1} (x - z) | \le \tfrac{1}{2}, i = 1, \ldots, n\} \\ S_{\mathrm{LS},z} &= \{x \in R^n \mid | c^T Q_{\hat{a}}^{-1} (x - z) | \le \tfrac{1}{2} c^T Q_{\hat{a}}^{-1} c, \forall c \in Z^n\} \end{aligned} \tag{16}$$

It can be shown that the two integer estimators $\check{a}_{\mathrm{R}}$ and $\check{a}_{\mathrm{LS}}$ are admissible and that their pull-in regions are integer-symmetric and convex. It then follows from Corollary 1 that both integer estimators are unbiased in the case that the 'float' solution is unbiased and normally distributed. And since the conditions of Corollary 2 apply, their success rates will get smaller in the presence of unaccounted biases.

We will now first describe the approach taken for constructing the upper bounds. To this end, we first note that the three pull-in regions of Eq. (16) are given as intersections of pairs of parallel half spaces. This implies that the pull-in region itself is a subset of any such single pair of intersecting parallel half spaces. The three upper bounds will therefore all be based on a subset of the type

$$U_0 = \{x \in R^n \mid | f^T x | \le \tfrac{1}{2}\} \tag{17}$$

in which the normal vector $f \in R^n$ still needs to be appropriately chosen. Since it follows from $x \sim N(b, Q_{\hat{a}})$ and $x \in U_0$, that

$$\frac{f^T (x - b)}{\| f \|_{Q_{\hat{a}}^{-1}}} \sim N(0, 1)$$

and

$$\frac{f^T (x - b)}{\| f \|_{Q_{\hat{a}}^{-1}}} \in \left[-\frac{1 + 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}}, \frac{1 - 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}} \right]$$

the upper bound follows as

$$P(x \in U_0) = \Phi\left(\frac{1 - 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}} \right) + \Phi\left(\frac{1 + 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}} \right) - 1 \tag{18}$$

For the three lower bounds, an alternative approach is taken. All three lower bounds will be based on an ellipsoidal subset of the type

$$L_0 = \{x \in R^n \mid x^T Q_{\hat{a}}^{-1} x \le \chi^2\} \tag{19}$$

in which χ^2 is a positive constant that still needs to be determined. The subset L_0 is thus chosen as the ellipsoid that just fits the pull-in region S_0. Once the constant χ^2 has been determined, the lower bound follows as

$$P(x \in L_0) = P(\chi^2(n, \| b \|_{Q_{\hat{a}}}^2) \le \chi^2) \tag{20}$$

with $\chi^2(n, \| b \|_{Q_{\hat{a}}}^2)$, the noncentral Chi-square distribution with n degrees of freedom and noncentrality parameter $\| b \|_{Q_{\hat{a}}}^2 = b^T Q_{\hat{a}}^{-1} b$. This result follows from the fact that if $x \sim N(b, Q_{\hat{a}})$, then $\| x \|_{Q_{\hat{a}}}^2 \sim \chi^2(n, \| b \|_{Q_{\hat{a}}}^2)$ (see e.g. Koch 1980). We are now in a position to formulate the three lower bounds and the three upper bounds.

Theorem 2 (*bounds on bias-affected success rates*). *The bias-affected ambiguity success rates of, respectively, 'rounding', 'bootstrapping' and 'integer LS', are bounded from above and below as:*

$$P(\chi^2(n, \| b \|_{Q_{\hat{a}}}^2) \le \chi^2) \ \le\ P_b(\check{a} = a) \ \le\ \Phi\left(\frac{1 - 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}} \right) + \Phi\left(\frac{1 + 2 f^T b}{2 \| f \|_{Q_{\hat{a}}^{-1}}} \right) - 1 \tag{21}$$

with

$$\begin{aligned} \chi^2 &= \frac{1}{4} \frac{1}{\max_i \sigma_{\hat{a}_i}^2} \quad f = c_i \quad \textit{for rounding} \\ \chi^2 &= \frac{1}{4} \frac{1}{\max_i \sigma_{\hat{a}_{i|I}}^2} \quad f = L^{-1} c_i \quad \textit{for bootstrapping} \\ \chi^2 &= \frac{1}{4} \min_{z \in Z^n / \{0\}} \| z \|_{Q_{\hat{a}}}^2 \quad f = \frac{1}{\| z \|_{Q_{\hat{a}}}^2} Q_{\hat{a}}^{-1} z \ \textit{for integer LS} \end{aligned}$$

and where c_i is one of the canonical unit vectors and $z \in Z^n$.

Proof. The proof for the upper bounds is rather straightforward. It follows once the normal vector f of the pair of parallel half spaces is given. For 'rounding', 'bootstrapping' and 'integer LS', we choose the three subsets respectively as $U_{\mathrm{R},0} = \{x \in R^n \mid | c_i^T x | \le \frac{1}{2}\}$, $U_{\mathrm{B},0} = \{x \in R^n \mid | c_i^T L^{-1} x | \le \frac{1}{2}\}$ and $U_{\mathrm{LS},0} = \{x \in R^n \mid | z^T Q_{\hat{a}}^{-1} x | \le \frac{1}{2} \| z \|_{Q_{\hat{a}}}^2, z \in Z^n\}$. The vector f is therefore given as, respectively, c_i, $L^{-1} c_i$ and $\frac{Q_{\hat{a}}^{-1} z}{\| z \|_{Q_{\hat{a}}}^2}$.

In order to prove the lower bounds, we make use of the following planes of support description of an ellipsoid (Teunissen and Kleusberg 1998, p 334):

$$\{x \in R^n \mid x^T Q_{\hat{a}}^{-1} x \leq \chi^2\} = \{x \in R^n \mid (g^T x)^2 \leq \chi^2 g^T Q_{\hat{a}} g,\ \forall\, g \in R^n\}$$

For ‘rounding’ this gives

$$\begin{aligned} L_{\mathrm{R},0} &= \{x \in R^n \mid (g^T x)^2 \leq \chi^2 g^T Q_{\hat{a}} g,\ \forall\, g \in R^n\} \\ &\subset \{x \in R^n \mid (c_i^T x)^2 \leq \chi^2 \sigma_{\hat{a}_i}^2,\ i = 1, \ldots, n\} \\ &\subset \{x \in R^n \mid (c_i^T x)^2 \leq \chi^2 \max_i \sigma_{\hat{a}_i}^2,\ i = 1, \ldots, n\} \\ &= S_{\mathrm{R},0} \text{ for } \chi^2 = \frac{1}{4} \frac{1}{\max_i \sigma_{\hat{a}_i}^2} \end{aligned}$$

The second line follows from replacing $\forall\, g \in R^n$ by the *finite set* of canonical unit vectors c_i, $i = 1, \ldots, n$, while the third line follows from replacing the variable upper bounds by the largest. For ‘bootstrapping’ we obtain in a similar way

$$\begin{aligned} L_{\mathrm{B},0} &= \{x \in R^n \mid (g^T x)^2 \leq \chi^2 g^T Q_{\hat{a}} g,\ \forall\, g \in R^n\} \\ &= \{x \in R^n \mid (c^T L^{-1} x)^2 \leq \chi^2 c^T D c,\ \forall\, c \in R^n\} \\ &\subset \{x \in R^n \mid (c_i^T L^{-1} x)^2 \leq \chi^2 \sigma_{\hat{a}_{i|I}}^2,\ i = 1, \ldots, n\} \\ &\subset \{x \in R^n \mid (c_i^T L^{-1} x)^2 \leq \chi^2 \max_i \sigma_{\hat{a}_{i|I}}^2,\ i = 1, \ldots, n\} \\ &= S_{\mathrm{B},0} \ \text{ for } \chi^2 = \frac{1}{4} \frac{1}{\max_i \sigma_{\hat{a}_{i|I}}^2} \end{aligned}$$

The second line follows from using the triangular factorization $Q_{\hat{a}} = LDL^T$ and from setting $g = L^{-T} c$. For ‘integer LS’ we find

$$\begin{aligned} L_{\mathrm{LS},0} &= \{x \in R^n \mid (g^T x)^2 \leq \chi^2 g^T Q_{\hat{a}} g,\ \forall\, g \in R^n\} \\ &= \{x \in R^n \mid (z^T Q_{\hat{a}}^{-1} x)^2 \leq \chi^2 z^T Q_{\hat{a}}^{-1} z,\ \forall\, z \in R^n\} \\ &\subset \{x \in R^n \mid (z^T Q_{\hat{a}}^{-1} x)^2 \leq \chi^2 z^T Q_{\hat{a}}^{-1} z,\ \forall\, z \in Z^n\} \\ &\subset \left\{ x \in R^n \mid (z^T Q_{\hat{a}}^{-1} x)^2 \leq \chi^2 \frac{\| z \|_{Q_{\hat{a}}}^4}{\min_{z \in Z^n / \{0\}} \| z \|_{Q_{\hat{a}}}^2}, \right. \\ &\qquad \left. \forall\, z \in Z^n \right\} \\ &= S_{\mathrm{LS},0} \ \text{ for } \chi^2 = \frac{1}{4} \min_{z \in Z^n / \{0\}} \| z \|_{Q_{\hat{a}}}^2 \end{aligned}$$

The second and third line follow from setting $g = Q_{\hat{a}}^{-1} z$ and restricting z to the space of integers. □

The following remarks can be made with respect to the above result. The lower bounds, when large enough, are useful to show that ambiguity resolution – despite the presence of biases – can still be expected to be successful, while the upper bounds, when small enough, are useful to show that successful ambiguity resolution will probably fail.

Note that the lower bounds for rounding and bootstrapping are dependent on the chosen ambiguity parametrization, whereas the lower bound for integer LS is not. The lower bounds for rounding and bootstrapping will become sharper when, respectively, the largest variance or largest conditional variance gets smaller. This can be achieved by means of the decorrelation process of the LAMBDA method. Also note that the three lower bounds depend only on the bias-to-noise ratio $\| b \|_{Q_{\hat{a}}}$. Different bias vectors having the same norm will therefore produce the same lower bound. This is in contrast with the success rate itself, which will generally also change when the direction of the bias vector changes.

There exists an ordering between the three lower bounds of Eq. (21). This is due to the inequalities

$$\frac{1}{\max_i \sigma_{\hat{a}_i}^2} \leq \frac{1}{\max_i \sigma_{\hat{a}_{i|I}}^2} \leq \min_{z \in Z^n / \{0\}} \| z \|_{Q_{\hat{a}}}^2 \tag{22}$$

This ordering implies that the lower bounds for rounding and bootstrapping are also lower bounds for integer LS, and that the lower bound for rounding is also a lower bound for bootstrapping.

The validity of the above two inequalities can be shown as follows. The first inequality follows, since $\sigma_{\hat{a}_i} \geq \sigma_{\hat{a}_{i|I}}, \forall\, i$, and thus $\max_j \sigma_{\hat{a}_j} \geq \sigma_{\hat{a}_{i|I}}, \forall\, i$. To prove the second inequality, let $\check{z}$ be the integer vector of minimum norm and let $Q_{\hat{a}} = LDL^T$ be the triangular factorization of the variance–covariance matrix. Then $D = \mathrm{diag}(\ldots, \sigma_{\hat{a}_{i|I}}^2, \ldots)$ and

$$\| \check{z} \|_{Q_{\hat{a}}}^2 = \sum_{i=1}^{n} \left(\frac{c_i^T L^{-1} \check{z}}{\sigma_{\hat{a}_{i|I}}} \right)^2$$

This shows, since L^{-1} is a *unit* lower triangular matrix, that $1/\sigma_{\hat{a}_{j|J}}^2 \leq \| \check{z} \|_{Q_{\hat{a}}}^2$, when $\check{z}_j \geq 1$ is the first nonzero entry of $\check{z}$ ($\check{z}_i = 0$ for $i < j$). The reciprocal value of the maximum conditional variance is therefore certainly less than or equal to the squared norm of $\check{z}$.

As to the upper bounds, note that we still have some freedom left in choosing them. The upper bounds for ‘rounding’ and ‘bootstrapping’ hold true for any vector c_i, while the upper bound for ‘integer LS’ holds true for any nonzero $z \in Z^n$. Since we would like the bounds to be as tight as possible, the upper bounds should be as small as possible. The upper bounds depend on the width and location of the integration interval. A large offset and a small width of the integration interval will produce a small upper bound. It follows from Eq. (18) that the width of the integration interval equals $\| f \|_{Q_{\hat{a}}^{-1}}^{-1}$, which is independent of the bias vector b. A good choice for f, when chosen to be independent of b, would therefore be the one that minimizes the width of the integration interval. For the three upper bounds, $\| f \|_{Q_{\hat{a}}^{-1}}$ will then equal $(\max_i \sigma_{\hat{a}_i})^{-1}$, $(\max_i \sigma_{\hat{a}_{i|I}})^{-1}$ and $\min_{z \in Z^n / \{0\}} \| z \|_{Q_{\hat{a}}}$, respectively. With these choices, the lower bounds become identical to the upper bounds when $n = 1$. In that case, $L_0 = \{x \in R \mid x^2 \leq \frac{1}{4}\}$ and $U_0 = \{x \in R \mid\mid x \mid\leq \frac{1}{2}\}$, and thus $L_0 = U_0$.

Finally note that the three integer estimators $\check{a}_{\mathrm{R}}$, $\check{a}_{\mathrm{B}}$ and $\check{a}_{\mathrm{LS}}$ become identical in the case that the ambiguity variance–covariance matrix is diagonal. The same holds true for the three lower bounds and also for the three upper bounds when f is appropriately chosen.

3.4 Simulating the success rate

Instead of using the results from the previous two theorems, we can choose the method of simulation to obtain an approximation of the success rate $P_b(\check{a} = a)$. We know that the 'float' solution is distributed as $\hat{a} \sim N(a + b, Q_{\hat{a}})$. We also know that the bias-affected success rate is independent of the mean a. Hence, in order to obtain the required success rate we may shift the distribution over a and restrict our attention to $N(b, Q_{\hat{a}})$, draw samples from it and use these samples to obtain the corresponding integer samples of the integer ambiguity estimator. Repeating this procedure a sufficient number of times allows us to built up the required frequency of correct integer estimation.

The following steps are followed when performing the simulation. First we start generating, using a random generator, n independent samples from the univariate standard normal distribution, say $s_1, \ldots, s_n$ from $N(0, 1)$. These samples are then collected in the vector $s = (s_1, \ldots, s_n)^T$ and transformed by means of $x = Gs$, where matrix G equals the Cholesky factor of the ambiguity variance–covariance matrix $Q_{\hat{a}}$, i.e. $Q_{\hat{a}} = GG^T$. Hence, x is now a sample from $N(0, Q_{\hat{a}})$. Adding the bias b to x, will then give the corresponding sample of $\hat{a} \sim N(b, Q_{\hat{a}})$. Using this sample, we can decide whether or not $\hat{a} \in S_0$. By repeating this process N number of times, we can count the number of times, N_a say, that the sample falls inside the pull-in region. An approximation to the success rate then follows as

$$P_b(\check{a} = a) \approx \frac{N_a}{N} \tag{23}$$

Successful ambiguity resolution can now be expected to be feasible when this probability is sufficiently close to 1. Note that this procedure requires an N-time verification of $\hat{a} \in S_0$. In the case of integer LS, with an inefficient implementation of the search, this becomes a very time consuming task when N is large. This shows that the simulation should not be based on the original DD ambiguities, but instead on the transformed ambiguities obtained by means of the LAMBDA method. Although the PMF of the transformed ambiguities differs from that of the DD ambiguities, their success rates will be the same.

In order to obtain an idea of how large N should be taken in the simulation, we consider the probability that N_0 out of N samples fall inside the pull-in region S_0. If the N samples are drawn independently from the normal distribution $N(b, Q_{\hat{a}})$, then this probability is governed by the binomial distribution and is given as

$$P(N_0) = \frac{N!}{(N - N_0)!N_0!} P_0^{N_0} (1 - P_0)^{N - N_0}$$

where we made use of the abbreviation $P_0 = P_b(\check{a} = 0)$. The mean (expectation) and variance (dispersion) of the relative frequency N_0/N follow therefore as

$$E\{N_0/N\} = P_0 \quad \text{and} \quad D\{N_0/N\} = P_0(1 - P_0)/N$$

Note that the first expression is in fact the motivation for using the relative frequency as an estimator for P_0, the probability of correct integer estimation. The second expression gives the precision of this estimator. It depends on both P_0 and N.

Using the above mean and variance we may now apply the Chebyshev inequality to obtain an upper bound on the probability that the relative frequency N_0/N differs more than ϵ from P_0. The corresponding Chebyshev inequality reads

$$P\left(\left|\frac{N_0}{N} - P_0\right| \geq \epsilon\right) \leq \frac{P_0(1 - P_0)}{N\epsilon^2} \tag{24}$$

The required number of samples N can be obtained by setting both ϵ and the upper bound to a small enough value. For instance, when the probability of correct integer estimation equals $P_0 = 1 - 10^{-3}$, an upper bound of 1% and a deviation of $\epsilon = 10^{-3}$ lead to a required number of samples of $N = 10^5$. This shows that in general a large number of samples are needed to obtain a sufficiently precise estimate of the probability of correct integer estimation. Instead of using the above Chebyshev inequality, we may also use the Gaussian approximation for the binomial distribution to obtain an estimate of the required number of samples, when N is large. This will usually give a somewhat less conservative estimate of N.

4 Summary

Carrier phase ambiguity resolution is the key to fast and high-precision GNSS positioning. Critical in the application of ambiguity resolution is the quality of the computed integer ambiguities. Unsuccessful ambiguity resolution, when passed unnoticed, will too often lead to unacceptable errors in the positioning results. In this contribution, the impact of unaccounted biases on the performance of carrier phase ambiguity resolution was evaluated for the first time. It was shown that biases in the 'float' solution generally reduce the ambiguity success rate. This reduction increases with the size of the bias along a fixed direction. Different approaches were discussed for evaluating the bias impact. The method of success rate simulation applies to any integer ambiguity estimator. In order to obtain an accurate approximation of the success rate, usually a high number of samples is needed. This implies that the method can become computationally demanding if an integer ambiguity estimator, such as the integer LS estimator, is implemented as a search. Application of the LAMBDA method will then help to ease the computational burden.

Simulation of the success rate is not needed for the bootstrapped estimator. It was shown that its bias-affected success rate could be given in closed form. It reads

$$P_b(\check{a}_{\mathrm{B}} = a) = \prod_{i=1}^{n} \left[\Phi\left(\frac{1 - 2\beta_i}{2\sigma_{\hat{a}_{i|I}}}\right) + \Phi\left(\frac{1 + 2\beta_i}{2\sigma_{\hat{a}_{i|I}}}\right) - 1 \right]$$

This success rate is very easy to compute. We only need the bias vector b together with the triangular factorization $Q_{\hat{a}} = LDL^T$. The entries of the diagonal matrix D provide the conditional variances and the triangular factor L, together with b, determines the bias components β_i.

For 'integer rounding' and 'integer LS' no such simple formula could be given. This is due to the relationship of their pull-in region's geometry with the nondiagonal structure of the ambiguity variance–covariance matrix. Instead of an exact expression, lower and upper bounds were given for their bias-affected success rates. For 'integer rounding' these bounds are given as

$$\begin{aligned} P\left(\chi^2(n, \| b \|^2_{Q_{\hat{a}}}) \le \frac{1}{4}\frac{1}{\max_i \sigma^2_{\hat{a}_i}}\right) \ &\le \ P_b(\check{a}_{\mathrm{R}} = a) \\ &\le \Phi\left(\frac{1-2b_i}{2\sigma_{\hat{a}_i}}\right) + \Phi\left(\frac{1+2b_i}{2\sigma_{\hat{a}_i}}\right) - 1 \end{aligned}$$

For 'integer LS' a similar approach was used for constructing the lower and upper bounds. They are given as

$$\begin{aligned} P\left(\chi^2(n, \| b \|^2_{Q_{\hat{a}}}) \le \frac{1}{4} \| \check{z} \|^2_{Q_{\hat{a}}}\right) \ &\le \ P_b(\check{a}_{\mathrm{LS}} = a) \\ &\le \Phi\left(\frac{1-2b_{\check{z}}}{2 \| \check{z} \|^{-1}_{Q_{\hat{a}}}}\right) + \Phi\left(\frac{1+2b_{\check{z}}}{2 \| \check{z} \|^{-1}_{Q_{\hat{a}}}}\right) - 1 \end{aligned}$$

in which $\check{z}$ denotes the nonzero integer minimizer of $\| z \|_{Q_{\hat{a}}}$ and $b_{\check{z}}$ equals the ratio of the orthogonal projection of b onto $\check{z}$ and the norm of $\check{z}$.

The lower bounds, when large enough, are useful to show that ambiguity resolution can still be expected tobe successful, while the upper bounds, when small enough, are useful to show that successful ambiguity resolution will probably fail.

References

Anderson TW (1955) The integral of a symmetric unimodal function over a symmetric convex set and some probability inequalities. Proc Am Math Soc 6: 170–176

Boon F, Ambrosius B (1997) Results of real-time applications of the LAMBDA method in GPS based aircraft landings. Proc KIS97, pp 339–345

Boon F, de Jonge PJ, Tiberius CCJM (1997) Precise aircraft positioning by fast ambiguity resolution using improved troposphere modelling. Proc ION GPS-97, vol 2, pp 1877–1884

Cox DB, Brading JDW (1999) Integration of LAMBDA ambiguity resolution with Kalman filter for relative navigation of spacecraft. Proc ION NTM 99, pp 739–745

de Jonge PJ, Tiberius CCJM (1996a) The LAMBDA method for integer ambiguity estimation: implementation aspects. LGR series no. 12, Delft Computing Centre, Delft

de Jonge PJ, Tiberius CCJM (1996b) Integer estimation with the LAMBDA method. In: Beutler G et al. (eds) Proc IAG Symp 115, GPS trends in terrestrial, airborne and spaceborne applications, Springer, Berlin Heidelberg New York, pp 280–284

de Jonge PJ, Tiberius CCJM, Teunissen PJG (1996) Computational aspects of the LAMBDA method for GPS ambiguity resolution. Proc ION GPS-96, pp 935–944

Han S (1995) Ambiguity resolution techniques using integer least-squares estimation for rapid static or kinematic positioning. Symp Satellite Navigation Technology: 1995 and beyond, Brisbane

Hofmann-Wellenhof B, Lichtenegger H, Collins J (1997) Global positioning system: theory and practice, 4th edn. Springer, Berlin Heidelberg New York

Jonkman NF (1998) Integer ambiguity estimation without the receiver–satellite geometry. LGR series no. 18, Delft Geodetic Computing Centre, Delft

Koch KR (1980) Parameterschaetzung und Hypothesetests in linearen Modellen. Dummler, Bonn

Leick A (1995) GPS satellite surveying, 2nd edn. John Wiley, New York

Parkinson B, Spilker JJ (eds) (1996) GPS: theory and applications, vols 1 and 2. AIAA, Washington DC

Peng HM, Chang FR, Wang LS (1999) Attitude determination using GPS carrier phase and compass data. Proc ION NTM 99, pp 727–732

Strang G, Borre K (1997) Linear algebra, geodesy, and GPS. Wellesley-Cambridge Press

Teunissen PJG (1993) Least-squares estimation of the integer GPS ambiguities. Invited lecture, Section IV Theory and Methodology, IAG General Meeting, Beijing, August 1993. Also in: LGR series no. 6, Delft Geodetic Computing Centre, Delft

Teunissen PJG (1999) An optimality property of the integer least-squares estimator. J Geod 73: 587–593

Teunissen PJG, Kleusberg A (eds) (1998) GPS for geodesy, 2nd edn. Springer, Berlin Heidelberg New York

Tiberius CCJM, de Jonge PJ (1995) Fast positioning using the LAMBDA method. Proc DSNS-95, paper 30

Tiberius CCJM, Teunissen PJG, de Jonge PJ (1997) Kinematic GPS: performance and quality control. Proc KIS97, pp 289–299