INTEGER APERTURE GNSS AMBIGUITY RESOLUTION

P.J.G. Teunissen
Department of Mathematical Geodesy and Positioning
Delft University of Technology
Kluyverweg 1
2629 HS Delft, The Netherlands
Fax: ++ 31 15 278 3711

Abstract
GNSS carrier phase ambiguity resolution is the key to fast and high- precision satellite positioning and navigation. It applies to a great variety of current and future models of GPS, modernized GPS and Galileo. In (*Teunissen*, 1998, 1999) we introduced the class of admissible integer (I) estimators and showed that the integer least-squares estimator is the optimal ambiguity estimator within this class. In (*Teunissen*, 2002a, b) we introduced the class of integer equivariant (IE) estimators and determined the best ambiguity estimator within this class. This best integer equivariant estimator is unbiased and of minimum variance.

In the present contribution we will introduce a third class of ambiguity estimators. This class of integer aperture (IA) estimators is larger than the I-class, but smaller than the IE-class, $I \subset IA \subset IE$. The IA-estimator is of a hybrid nature since its outcome may be integer-valued or real-valued. We also give a probabilistic description of IA-estimators. This is needed in order to be able to propagate the inherent uncertainty in the data rigorously and to give a proper probabilistic evaluation of the final result. The framework of IA-estimation also incorporates the important problem of ambiguity discernibility. By setting the size and shape of the integer aperture pull-in region of an IA-estimator, the user has control over the fail- rate of the estimator and thus also over the amount of discernibility.

1 Introduction

As our point of departure we take the following system of linear observation equations

$$E\{y\} = Aa + Bb \ , \ a \in Z^n \ , \ b \in R^p \tag{1}$$

with $E\{.\}$ the mathematical expectation operator, y the m-vector of observables, a the n-vector of unknown integer parameters and b the p-vector of unknown real-valued parameters. The $m \times (n+p)$ design matrix (A, B) is assumed to be of full rank. All the linear(ized) GNSS models can in principle be cast in the above frame of observation equations. The data vector y will then usually consist of the 'observed minus computed' single- or dual- frequency double-difference (DD) phase and/or pseudorange (code) observations accumulated over all observation epochs. The entries of vector a are then the DD carrier phase ambiguities, expressed in units of cycles rather than range, while the entries of the vector b will consist of the remaining unknown parameters, such as for instance baseline components (coordinates) and possibly atmospheric delay parameters (troposphere, ionosphere).

The procedure which is usually followed for solving the GNSS model can be divided into three steps. In the *first* step one simply discards the integer constraints $a \in Z^n$ and performs a standard least-squares (LS) adjustment. As a result one obtains the LS-estimators of a and b as

$$\left\{ \begin{array}{lcl} \hat{a} & = & (\bar{A}^T Q_y^{-1} \bar{A})^{-1} \bar{A}^T Q_y^{-1} y \\ \hat{b} & = & (\bar{B}^T Q_y^{-1} \bar{B})^{-1} \bar{B}^T Q_y^{-1} y \end{array} \right. \tag{2}$$

with Q_y the vc-matrix of the observables, $\bar{A} = P_B^{\perp} A$, $\bar{B} = P_A^{\perp} B$, and the two orthogonal projectors $P_B^{\perp} = I_m - B(B^T Q_y^{-1} B)^{-1} B^T Q_y^{-1}$ and $P_A^{\perp} = I_m - A(A^T Q_y^{-1} A)^{-1} A^T Q_y^{-1}$. This solution is usually referred to as the 'float' solution.

In the *second* step the 'float' estimator $\hat{a}$ is used to compute an improved estimator which in some pre-defined sense incorporates the integerness of the ambiguities. This estimator is given as

$$\hat{a}_S = S(\hat{a}) \tag{3}$$

where $S : R^n \mapsto R^n$. This improved ambiguity estimator is then used in the final and *third* step to adjust the 'float' estimator $\hat{b}$. As a result one obtains the so-called 'fixed' estimator of b as

$$\check{b} = \hat{b} - Q_{\hat{b}\hat{a}} Q_{\hat{a}}^{-1} (\hat{a} - \hat{a}_S) \tag{4}$$

in which $Q_{\hat{a}}$ denotes the vc-matrix of $\hat{a}$ and $Q_{\hat{b}\hat{a}}$ denotes the covariance matrix of $\hat{b}$ and $\hat{a}$. This 'fixed' estimator can alternatively be expressed as $\check{b} = (B^T Q_y^{-1} B)^{-1} B^T Q_y^{-1} (y - A\hat{a}_S)$.

The above three-step procedure is still ambiguous in the sense that it depends on which mapping S is chosen. Different choices for S will lead to different ambiguity estimators $\hat{a}_S$ and thus also to different baseline estimators $\check{b}$. One can therefore now think of constructing maps which possess certain desirable properties. In earlier contributions the author introduced the class of admissible integer (I) estimators and the class of integer equivariant (IE) estimators. In the present contribution the class of integer aperture (IA) estimators will be introduced. This class is larger than the I-class, but smaller than the IE-class. It will also be shown how the probabilistic performance of the IA-estimators can be evaluated.

2 Integer estimation and integer equivariant estimation

2.1 The class of integer estimators

If one requires the output of the map S to be integer, $S : R^n \mapsto Z^n$, then S will not be one-to-one due to the discrete nature of Z^n. Instead it will be a many-to-one map. This implies that different real-valued vectors will be mapped to one and the same integer vector. One can therefore assign a subset $S_z \subset R^n$ to each integer vector $z \in Z^n$:

$$S_z = \{x \in R^n \mid z = S(x)\}, \quad z \in Z^n \tag{5}$$

The subset S_z contains all real-valued vectors that will be mapped by S to the same integer vector $z \in Z^n$. This subset is referred to as the *pull-in region* of z. It is the region in which all vectors are pulled to the same integer vector z.

Since the pull-in regions define the integer estimator completely, one can define classes of integer estimators by imposing various conditions on the pull-in regions. One such class was introduced in (*Teunissen*, 1998a) as follows.

Definition 1 (*Integer estimators*)
The mapping $\check{a} = S(\hat{a})$, with $S : R^n \mapsto Z^n$, is said to be an integer estimator if its pull-in regions satisfy

$$\begin{array}{ll} (i) & \bigcup_{z \in Z^n} S_z = R^n \\ (ii) & Int(S_{z_1}) \bigcap Int(S_{z_2}) = \emptyset, \ \ \forall z_1, z_2 \in Z^n, z_1 \neq z_2 \\ (iii) & S_z = z + S_0, \ \ \forall z \in Z^n \end{array}$$

This definition is motivated as follows. Each one of the above three conditions describe a property of which it seems reasonable that it is possessed by an arbitrary integer 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 needed in order to be able to map any float solution $\hat{a} \in R^n$ to Z^n, while the absence of overlaps is needed 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. This is permitted if we assume to have zero probability that $\hat{a}$ lies on one of the boundaries. This will be the case when the probability density function (pdf) of $\hat{a}$ is continuous.

The third and last condition of the definition follows from the requirement that $S(x+z) = S(x) + z, \forall x \in R^n, z \in Z^n$. Also this condition is a reasonable one to ask for. It states that when the float solution $\hat{a}$ is perturbed by $z \in Z^n$, the corresponding integer solution is perturbed by the same amount. This property allows one to apply the *integer remove-restore* technique: $S(\hat{a}-z)+z = S(\hat{a})$. It therefore allows one to work with the fractional parts of the entries of $\hat{a}$, instead of with its complete entries.

Using the pull-in regions, one can give an explicit expression for the corresponding integer estimator $\check{a}$. It reads

$$\check{a} = \sum_{z \in Z^n} z s_z(\hat{a}) \quad \text{with} \quad s_z(\hat{a}) = \begin{cases} 1 & \text{if} \quad \hat{a} \in S_z \\ 0 & \text{if} \quad \hat{a} \notin S_z \end{cases} \tag{6}$$

Note that the $s_z(\hat{a})$ can be interpreted as weights, since $\sum_{z \in Z^n} s_z(\hat{a}) = 1$. The integer estimator $\check{a}$ is therefore equal to a weighted sum of integer vectors with binary weights.

The three best known integer estimators are integer rounding, integer bootstrapping and integer least-squares. The simplest way to obtain an integer vector from the real-valued float solution is to round each of the entries of $\hat{a}$ to its nearest integer. The corresponding integer estimator reads therefore

$$\check{a}_R = ([\hat{a}_1], \ldots, [\hat{a}_n])^T \tag{7}$$

where '[.]' denotes rounding to the nearest integer. The pull-in region of this integer estimator equals the multivariate version of the unit- square.

Another relatively simple integer ambiguity estimator is the bootstrapped estimator. The bootstrapped estimator can be seen as a generalization of the previous estimator. It still makes use of integer rounding, but it also takes some of the correlation between the ambiguities into account. The bootstrapped estimator follows from a sequential conditional least- squares adjustment and it is computed as follows. If n ambiguities are available, one starts with the first ambiguity $\hat{a}_1$, and rounds 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 considered. The entries of the bootstrapped estimator $\check{a}_B = (\check{a}_{B,1}, \ldots, \check{a}_{B,n})^T \in Z^n$ are thus given as

$$\begin{cases} \check{a}_{B,1} & = & [\hat{a}_1] & & \\ \check{a}_{B,2} & = & [\hat{a}_{2|1}] & = & [\hat{a}_2 - \sigma_{21}\sigma_1^{-2}(\hat{a}_1 - \check{a}_{B,1})] \\ & \vdots & & & \\ \check{a}_{B,n} & = & [\hat{a}_{n|N}] & = & [\hat{a}_n - \sum_{j=1}^{n-1} \sigma_{n,j|J}\sigma_{j|J}^{-2}(\hat{a}_{j|J} - \check{a}_{B,j})] \end{cases} \tag{8}$$

where $\sigma_{i,j|J}$ denotes the covariance between $\hat{a}_i$ and $\hat{a}_{j|J}$, and $\sigma_{j|J}^2$ is the variance of $\hat{a}_{j|J}$. The shorthand notation $\hat{a}_{i|I}$ stands for the ith least-squares ambiguity obtained through a conditioning on the previous $I = \{1, \ldots, (i-1)\}$ sequentially rounded ambiguities. The pull-in region of the bootstrapped estimator is the multivariate version of a parallellogram. Note that the bootstrapped pull-in regions reduce to multivariate unit- cubes in case the vc-matrix is a diagonal matrix. Bootstrapping reduces namely to rounding in the absence of any correlation between the ambiguities.

Also note that the bootstrapped estimator is not unique. The outcome of bootstrapping and its performance depend on the chosen ambiguity parametrization. Thus although the principle of bootstrapping remains the same, every choice of ambiguity parametrization has its own bootstrapped estimator. Bootstrapping of DD-ambiguities, for instance, will generally perform poorly due to the high correlation and poor precision of DD ambiguities when short observation time spans are used. One should therefore make use of an appropriate parametrization when using bootstrapping. This can be done by applying the decorrelating Z- transformation of the LAMBDA (Least-squares AMBiguity Decorrelation Adjustment) method. When this transformation is applied, one works with the more precise and decorrelated ambiguity vector $\hat{z} = Z\hat{a}$, instead of with the original ambiguity vector $\hat{a}$. For more information on the LAMBDA method, we refer to (*Teunissen*, 1993) or to the textbooks (*Hofmann-Wellenhof et al.*, 2002), (*Strang and Borre*, 1997), (*Teunissen and Kleusberg*, 1998), (*Misra and Enge*, 2002) and (*Seeber*, 2003).

The integer least-squares estimator is defined as

$$\check{a}_{LS} = \arg\min_{z \in Z^n} \| \hat{a} - z \|^2_{Q_{\hat{a}}} \tag{9}$$

In contrast to integer rounding and integer bootstrapping, an integer search is needed to compute $\check{a}_{LS}$. The ILS procedure is mechanized in the LAMBDA method, which is currently one of the most applied methods for GNSS carrier phase ambiguity resolution. In (*Teunissen*, 1999) it has been shown that the integer least-squares estimator is optimal in the sense that it has the highest possible success-rate of all integer estimators. Its pull-in regions are convex, symmetric sets of volume 1, which satisfy the conditions of Definition 1. They are hexagons in the two-dimensional case. In higher dimensions they are constructed from at most $2^n - 1$ pairs of intersecting half-spaces.

2.2 The class of integer equivariant estimators

One may wonder what happens if the conditions of Definition 1 are relaxed. Would it then still be possible to find a sensible ambiguity estimator? In order to answer this question the class of *integer equivariant* (IE) estimators was introduced in (*Teunissen*, 2002a). This class is larger than the class of integer estimators and it is defined as follows.

Definition 2 (*Integer equivariant estimators*)
The estimator $\hat{\theta}_{IE} = F_\theta(\hat{a})$, with $F_\theta : R^n \mapsto R$, is said to be an integer equivariant estimator of $\theta = l^T a$ if

$$F_\theta(x + z) = F_\theta(x) + l^T z \ , \ \forall x \in R^n, z \in Z^n \tag{10}$$

This definition was motivated by the fact that of the conditions of Definition 1 one should at least retain the property that the *integer remove-restore* principle applies. It will be clear that integer (I) estimators are also IE-estimators. Simply check that the above condition is indeed fulfilled by the estimator $\check{\theta} = l^T \check{a}$. The converse, however, is not necessarily true. The class of IE-estimators is therefore indeed a larger class than the class of I-estimators.

The class of IE-estimators is also a larger class than the class of linear unbiased estimators, assuming that the float solution is unbiased. Let $F_\theta^T \hat{a}$, for some $F_\theta \in R^n$, be the linear estimator of $\theta = l^T a$. For it to be unbiased one needs, using $E\{\hat{a}\} = a$, that $F_\theta^T E\{\hat{a}\} = l^T a$, $\forall a \in R^n$ holds true, or that $F_\theta = l$. But this is equivalent to stating that $F_\theta^T(\hat{a} + a) = F_\theta^T \hat{a} + l^T a$, $\forall \hat{a} \in R^n, a \in R^n$. Comparison with (**??**) shows that the condition of linear unbiasedness is more restrictive than the condition of integer equivariance. The class of linear unbiased estimators is therefore a subset of the class of integer equivariant estimators.

This result implies that IE-estimators exist which are unbiased. Thus, if one denotes the class of IE-estimators as IE, the class of unbiased estimators as U, the class of unbiased IE-estimators as IEU, the class of unbiased integer estimators as IU, and the class of linear unbiased estimators as LU, one may summarize their relationships as: $IEU = IE \cap U \neq \emptyset$, $LU \subset IEU$ and $IU \subset IEU$

Having defined the class of IE-estimators one may now look for an IE- estimator which is 'best' in a certain sense. We use the mean squared error (MSE) as our criterion of 'best' and denote the *best integer equivariant* (BIE) estimator as $\hat{\theta}_{BIE}$. The best integer equivariant estimator of $\theta = l^T a$ is therefore defined as

$$\hat{\theta}_{BIE} = \arg \min_{F_\theta \in IE} E\{(F_\theta(\hat{a}) - \theta)^2\} \tag{11}$$

in which IE stands for the class of IE-estimators. The minimization is thus taken over all integer equivariant functions that satisfy the condition of Definition 2.

The solution to this optimization problem is given in (*Teunissen*, 2002b). The reason why we choose the MSE-criterion is twofold. First, it is a well-known probabilistic criterion for measuring the closeness of an estimator to its target value, in our case $\theta = l^T a$. Second, the MSE-criterion is also often used as measure for the quality of the float solution itself. Note that the BIE-estimator is *always* better than the float solution. Afterall the float solution is an IE-estimator as well. It should be kept in mind however that the MSE-criterion is a weaker criterion than the probabilistic criterion of maximizing the success-rate as used in the previous section.

3 Integer aperture estimation

The two classes of ambiguity estimators discussed in the previous section are related as $I \subset IE$. That is, integer estimators are integer equivariant, but integer equivariant estimators are not necessarily integer. We will now introduce a third class of ambiguity estimators. It will be referred to as the class of *integer aperture* (IA) estimators. This class will be larger than the I-class, but smaller than the IE-class, $I \subset IA \subset IE$. Whereas the IE-class was defined by dropping two of the three conditions of Definition 1, the IA-class will be defined by dropping only one of the three conditions, namely the condition that the pull-in regions should cover R^n completely. We will therefore allow the pull-in regions of the IA-estimators to have gaps.

In order to introduce the new class of ambiguity estimators from first principles, let $\Omega \subset R^n$ be the region of R^n for which $\hat{a}$ is mapped to an integer if $\hat{a} \in \Omega$. It seems reasonable to ask of the region Ω that it has the property that if $\hat{a} \in \Omega$ then also $\hat{a} + z \in \Omega$, for all $z \in Z^n$. If this property would not hold, then float solutions could be mapped to integers whereas their fractional parts would not. We thus require Ω to be translational invariant with respect to an arbitrary integer vector: $\Omega + z = \Omega$, for all $z \in Z^n$. Knowing Ω is however not sufficient for defining our estimator. Ω only determines whether or not the float solution is mapped to an integer, but it does not tell us yet to which integer the float solution is mapped. We therefore define

$$\Omega_z = \Omega \cap S_z \ , \ \forall z \in Z^n \tag{12}$$

where S_z is a pull-in region satisfying the conditions of Definition 1. Then

$$\begin{array}{ll} (i) & \bigcup_z \Omega_z = \bigcup_z (\Omega \bigcap S_z) = \Omega \bigcap (\bigcup_z S_z) = \Omega \bigcap R^n = \Omega \\ (ii) & \Omega_{z_1} \bigcap \Omega_{z_2} = (\Omega \bigcap \Omega_{z_1}) \bigcap (\Omega \bigcap \Omega_{z_2}) = \Omega \bigcap (S_{z_1} \bigcap S_{z_2}) = \emptyset, \ \ \forall z_1, z_2 \in Z^n, z_1 \neq z_2 \\ (iii) & \Omega_0 + z = (\Omega \bigcap S_0) + z = (\Omega + z) \bigcap (S_0 + z) = \Omega \bigcap S_z = \Omega_z, \ \ \forall z \in Z^n \end{array}$$

This shows that the subsets $\Omega_z \subset S_z$ satisfy the same conditions as those of Definition 1, be it that R^n has now been replaced by $\Omega \subset R^n$. Hence, the mapping of the IA-estimator can now be defined as follows. The IA-estimator maps the float solution $\hat{a}$ to the integer vector z when $\hat{a} \in \Omega_z$ and it maps the float solution to itself when $\hat{a} \notin \Omega$. The class of IA-estimators can therefore be defined as follows.

Definition 3 (*Integer aperture estimators*)
Integer aperture estimators are defined as

$$\hat{a}_{IA} = \hat{a} + \sum_{z \in Z^n} (z - \hat{a}) \omega_z(\hat{a}) \tag{13}$$

with $\omega_z(x)$ the indicator function of $\Omega_z = \Omega \cap S_z$ and $\Omega \subset R^n$ translational invariant.

Note that an IA-estimator is indeed also an IE-estimator, just like an I-estimator is. There is also resemblence between an IA-estimator and an I-estimator. Since the indicator functions $s_z(x)$ of the pull-in regions S_z sum up to unity, $\sum_{z \in Z^n} s_z(x) = 1$, the I-estimator (**??**) may be written as

$$\check{a} = \hat{a} + \sum_{z \in Z^n} (z - \hat{a}) s_z(x) \tag{14}$$

Comparing this expression with that of (**??**) shows that the difference between the two estimators lies in their binary weights, $s_z(x)$ versus $\omega_z(x)$. Since the $s_z(x)$ sum up to unity for all $x \in R^n$, the outcome of an I-estimator will always be integer. This is not true for an IA-estimator, since the binary weights $\omega_z(x)$ do not sum up to unity for all $x \in R^n$. The IA-estimator is therefore an hybrid estimator having as outcome either the real-valued float solution $\hat{a}$ or an integer solution. The IA-estimator returns the float solution if $\hat{a} \notin \Omega$ and it will be equal to z when $\hat{a} \in \Omega_z$. Note, since Ω is the collection of all $\Omega_z = \Omega_0 + z$, that the IA-estimator is completely determined once Ω_0 is known. Thus $\Omega_0 \subset S_0$ plays the same role for the IA-estimators as S_0 does for the I-estimators. By changing the size and shape of Ω_0 one changes the outcome of the IA-estimator. The subset Ω_0 can therefore be seen as an adjustable pull-in region with two limiting cases. The limiting case in which Ω_0 is empty and the limiting case when Ω_0 equals S_0. In the first case the IA-estimator becomes identical to the float solution $\hat{a}$, and in the second case the IA-estimator becomes identical to an I-estimator. The subset Ω_0 therefore determines the *aperture* of the pull-in region.

4 Evaluation of IA-estimators

In order to be able to evaluate the performance of an IA-estimator, one needs its probability density function (pdf). It can be obtained from the pdf of the float solution if one discriminates between the following two disjunct cases: the case that $\hat{a} \notin \Omega_z$ and the case that $\hat{a} \in \Omega_z$ for some $z \in Z^n$. Since $\hat{a}_{IA} = \hat{a}$ if $\hat{a} \notin \Omega_z$, the pdf of $\hat{a}_{IA}$ will equal the pdf of $\hat{a}$ when the first case applies. For the second case one has $\hat{a}_{IA} = z$. The probability that this occurs, $P(\hat{a}_{IA} = z)$, equals the integral of the pdf of $\hat{a}$ over the region Ω_z. Hence, for the second case the pdf of $\hat{a}_{IA}$ equals a weighted train of impulse functions, with weights $P(\hat{a}_{IA} = z)$. The complete pdf of $\hat{a}_{IA}$ follows then by combining the two disjunct cases. The result is given in the following Theorem.

Theorem (*The pdf of an IA-estimator*)
Let $f_{\hat{a}}(x)$ be the pdf of the float solution $\hat{a}$ and let $\omega_z(x)$ be the indicator function which defines the IA-estimator $\hat{a}_{IA}$. Then the pdf of $\hat{a}_{IA}$ is given as

$$f_{\hat{a}_{IA}}(x) = f_{\hat{a}}(x) \bar{\omega}(x) + \int_{R^n} f_{\check{\epsilon}}(v) \omega_0(v) dv \delta(x - z) \tag{15}$$

where $\delta(x)$ denotes the impulse function, $\bar{\omega}(x) = \left(1 - \sum_{z \in Z^n} \omega_z(x)\right)$ is the indicator function of $R^n \setminus \Omega$ and $f_{\check{\epsilon}}(x) = \sum_{z \in Z^n} f_{\hat{a}}(x + z) s_0(x)$ is the pdf of the residual $\check{\epsilon} = \hat{a} - \check{a}$.

Note that the pdf of an IA-estimator is discontinuous. This is a consequence of the hybrid nature of the estimator. Having defined the class of IA-estimators in which each estimator is uniquely defined by its aperture pull-in region Ω_0, one can now design one's own IA-estimator and evaluate its performance by using the pdf $f_{\hat{a}_{IA}}(x)$. As an example one may consider the first moment of an IA-estimator and study the conditions under which the IA-estimator becomes unbiased. The following corollary gives the result.

Corollary (*The mean of an IA-estimator*)
Let $f_{\hat{a}}(x)$ be the pdf of the float solution $\hat{a}$ and let Ω_0 be the aperture pull-in region which uniquely defines the IA-estimator. Then

$$E\{\hat{a}_{IA}\} = E\{\hat{a}\} - \int_{\Omega_0} x f_{\check{\epsilon}}(x) dx \tag{16}$$

with $f_{\check{\epsilon}}(x) = \sum_{z \in Z^n} f_{\hat{a}}(x+z) s_0(x)$ the pdf of the ambiguity residual $\check{\epsilon} = \hat{a} - \check{a}$.

Proof: Substitution of (**??**) into $E\{\hat{a}_{IA}\} = \int_{R^n} x f_{\hat{a}_{IA}}(x) dx$ gives

$$E\{\hat{a}_{IA}\} = \int_{R^n} x f_{\hat{a}}(x) dx + \sum_{z \in Z^n} \int_{\Omega_z} (z - x) f_{\hat{a}}(x) dx$$

from which the result follows when using the Theorem and a change of variables in the second term of the sum. *End of proof.*

The above result shows that an IA-estimator is unbiased when Ω_0 and $f_{\check{\epsilon}}(x)$ are both symmetric with respect to the origin.

In order to evaluate the performance of an IA-estimator as to whether it produces the correct integer outcome $a \in Z^n$, it is helpful to classify its possible outcomes. An IA-estimator can produce one of the following three outcomes

$$\hat{a}_{IA} = \begin{cases} a \in Z^n & \text{(correct integer)} \\ z \in Z^n \setminus \{a\} & \text{(incorrect integer)} \\ \hat{a} \in R^n \setminus Z^n & \text{(no integer)} \end{cases} \tag{17}$$

A correct integer outcome may be considered a *success*, an incorrect integer outcome a *failure*, and an outcome where no correction at all is given to the float solution as indeterminate or *undecided*. The probability of success, the *success-rate*, equals the integral of $f_{\hat{a}_{IA}}(x)$ over Ω_a, whereas the probability of failure, the *fail-rate*, equals the probability of $f_{\hat{a}_{IA}}(x)$ over $\Omega \setminus \Omega_a$. The respective probabilities are therefore given as

$$\begin{cases} P_S &= P(\hat{a}_{IA} = a) &= \int_{\Omega_a} f_{\hat{a}_{IA}}(x) dx &= \int_{\Omega_a} f_{\hat{a}}(x) dx \quad \text{(success)} \\ P_F &= \sum_{z \neq a} P(\hat{a}_{IA} = z) &= \sum_{z \neq a} \int_{\Omega_z} f_{\hat{a}_{IA}}(x) dx &= \sum_{z \neq a} \int_{\Omega_z} f_{\hat{a}}(x) dx \quad \text{(failure)} \\ P_U &= P(\hat{a}_{IA} = \hat{a}) &= 1 - \int_{\Omega} f_{\hat{a}_{IA}}(x) dx &= 1 - P_S - P_F \quad \text{(undecided)} \end{cases} \tag{18}$$

Note that these three probabilities are completely governed by $f_{\hat{a}}(x)$, the pdf of the float solution, and by Ω_0, the aperture pull-in region which uniquely defines the IA-estimator. Hence, as it was the case with the mean of $\hat{a}_{IA}$, one can proceed with the evaluation of IA-estimators once this information is available.

Depending on the type of IA-estimator one is considering, the above integrals for computing the success-rate and the fail-rate may be difficult to evaluate exactly. Whether or not an exact evaluation is possible depends to a large extent on the complexity of the geometry of the aperture pull-in region Ω_0. In a forthcoming contribution an IA-estimators will be introduced for which an exact computation of P_S and P_F is possible (*Teunissen*, 2003). For others however such an exact evaluation may not be feasible. In that case one has to use the method of simulation. If one may assume that the float solution is Gaussian distributed as $\hat{a} \sim \mathrm{N}(a, Q_{\hat{a}})$, the simulation of the fail-rate and the success-rate goes as follows. Since the shape of the Gaussian distribution is independent of the mean a, also P_S and P_F are independent of a. Hence, one may restrict attention to $\mathrm{N}(0, Q_{\hat{a}})$, draw samples from it and use these samples to obtain good approximations to both P_S and P_F.

As a first step one generates, using a random generator, n independent samples from the univariate standard normal distribution $\mathrm{N}(0,1)$, say $s_1,\ldots,s_n$. These samples are then collected in the vector $s=(s_1,\ldots,s_n)^T$ and transformed by means of $\hat{a}=Gs$, where matrix G equals the Cholesky factor of $Q_{\hat{a}}$, i.e. $Q_{\hat{a}}=GG^T$. Hence, $\hat{a}$ is now a sample from $\mathrm{N}(0,Q_{\hat{a}})$. This sample is then used as input for the IA-estimator. The outcome of the IA-estimator is then correct if the output equals the zero vector, it is incorrect if the output equals a nonzero integer vector and the outcome is undecided if the output equals the input. The first case corresponds with $\hat{a}\in\Omega_0$, the second case with $\hat{a}\in\Omega_z$ for some $z\in Z^n\setminus\{0\}$, and the third case with $\hat{a}\notin\Omega_z,\forall z\in Z^n$. By repeating this process an N-number of times one can count how often the zero vector is given as solution, say N_0-times, and how often a particular nonzero integer vector is given as solution, say N_z-times. An approximation to the required success-rate and the required fail-rate follows then from the relative frequencies as

$$P_S\approx\frac{N_0}{N}\quad,\quad P_F\approx\frac{\sum_{z\neq 0}N_z}{N} \tag{19}$$

Note that this procedure requires the evaluation whether or not the generated sample $\hat{a}$ resides in one of the aperture pull-in regions. Since $\Omega_z=\Omega\cap S_z$, with S_z the pull-in region of the chosen integer estimator, this evaluation is done in two steps. First the integer vector corresponding to $\hat{a}$ is computed, say $\check{a}$. Depending on the choice of S_z, this could be based on integer rounding, integer bootstrapping, integer least-squares or any other admissible integer estimator. Then in the second step the residual $\check{\epsilon}=\hat{a}-\check{a}$ is used to verify whether or not $\check{\epsilon}\in\Omega_0$. Since this procedure has to be repeated N-times, it is of importance that the integer solution $\check{a}$ can be computed as efficiently as possible. For the case of GPS this implies that one should not use the original ambiguities, but instead the transformed and decorrelated ambiguities as obtained with the LAMBDA method. As to the choice of N, we refer to e.g. (*Teunissen*, 1998b). For more advanced methods of approximating the integrals of P_S and P_F using Monte-Carlo or other methods, we refer to e.g. (*Evans and Schwartz*, 2000).

It can also be shown that some of the popular discernibility procedures currently in place for carrier phase ambiguity resolution are in fact applications of different IA-estimators. This holds true for e.g. the ratio-test, the difference-test and the projector-test. In fact by setting the size and shape of the aperture pull-in region of an IA-estimator one has control over one of the three probabilities, P_F, P_S or P_U, and thereby one can exercise one's own influence on the amount of discernibilty. By establishing this fact, one is thus now for the first time able to compare these different procedures using the framework of IA-estimation and to systematically evaluate their probabilistic properties and performance by using the pdf $f_{\hat{a}_{IA}}(x)$.

5 References

Evans, M. and T. Schwartz (2000): *Approximating integrals via Monte Carlo and deterministic methods*, Oxford University Press, 2000.

Hofmann-Wellenhof, B., H. Lichtenegger, J. Collins (2001): *Global Positioning System: Theory and Practice*. 5th edition. Springer Verlag.

Leick, A. (1995): *GPS Satellite Surveying*. 2nd edition, John Wiley, New York.

Misra, P., P. Enge (2001): *Global Positioning System: Signals, Measurements, and Performance*. Ganga-Jamuna Press, 2001.

Parkinson, B., J.J. Spilker (eds) (1996): *GPS: Theory and Applications*, Vols 1 and 2, AIAA, Washington DC.

Seeber, G. (2003): *Satellite Geodesy*, De Gruyter, 2nd ed.

Strang, G., K. Borre (1997): *Linear Algebra, Geodesy, and GPS*, Wellesley-Cambridge Press.

Teunissen, P.J.G., A. Kleusberg (eds) (1998): *GPS for Geodesy*, 2nd enlarged edition, Springer Verlag.

Teunissen, P.J.G. (1993): Least-squares estimation of the integer GPS ambiguities. Invited Lecture, Section IV Theory and Methodology, IAG General Meeting, Beijing, China, August 1993. Also in: *LGR Series*, No. 6, Delft Geodetic Computing Centre.

Teunissen, P.J.G. (1998a): A class of unbiased integer GPS ambiguity estimators. *Artificial Satellites*, Vol. 33, No. 1, pp. 3-10.

Teunissen, P.J.G. (1998b): On the integer normal distribution of the GPS ambiguities, *Artificial Satellites*, Vol. 33, No. 2, pp. 49-64.

Teunissen, P.J.G. (1999): A theorem on maximizing the probability of correct integer estimation. *Artificial Satellites*, Vol. 34, No. 1, pp. 3-10.

Teunissen, P.J.G. (2002a): A new class of GNSS ambiguity estimators. *Artificial Satellites*, Vol. 37, No. 4, pp. 111-120.

Teunissen, P.J.G. (2002b): A Gauss-Markov-like theorem for integer GNSS ambiguities. *Artificial Satellites*, Vol. 37, No. 4, pp. 121-128.

Teunissen, P.J.G. (2003): A carrier-phase ambiguity estimator with easy- to-evaluate fail-rate. *Artificial Satellites*, Vol. 38, No. 4, in print.