An invariant upperbound for the GNSS bootstrapped ambiguity success-rate

P.J.G. Teunissen

Department of Mathematical Geodesy and Positioning, Delft University of Technology, Kluyverweg 1, 2629 HS Delft

Received: 2 June 2003 / Accepted: 16 June 2003

Abstract. Carrier phase ambiguity resolution is the key to fast and high precision GPS 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. The success or failure of carrier phase ambiguity resolution can be predicted by means of the probability of correct integer estimation, also referred to as the ambiguity success-rate. The upperbound is easy to compute and it is invariant for the class of admissible ambiguity transformations. In this contribution we prove an upperbound for the bootstrapped success-rate. The upperbound is easy to compute and it is invariant for the class of admissible ambiguity transformations.

Keywords GNSS, integer bootstrapping, ADOP, ambiguity success-rate

1 Introduction

There are many ways of computing an integer ambiguity vector \( \hat{a} \in \mathbb{Z}^n \) from its real-valued counterpart \( \hat{a} \in \mathbb{R}^n \), also referred to as the 'float' solution. To each such method belongs a mapping \( S : \mathbb{R}^n \mapsto \mathbb{Z}^n \) from the \( n \)-dimensional space of real numbers to the \( n \)-dimensional space of integers. Due to the discrete nature of \( \mathbb{Z}^n \), the map \( S \) will not be one-to-one, but instead a many-to-one map. This implies that different real-valued ambiguity vectors will be mapped to the same integer vector. One can therefore assign a subset \( S_z \subseteq \mathbb{R}^n \) to each integer vector \( z \in \mathbb{Z}^n \):

\[
S_z = \{ x \in \mathbb{R}^n \mid z = S(x) \}, \quad z \in \mathbb{Z}^n
\]  

(1)

The subset \( S_z \) contains all real-valued ambiguity vectors that will be mapped by \( S \) to the same integer vector \( z \in \mathbb{Z}^n \). This subset is referred to as the pull-in region of \( z \). It is the region in which all ambiguity 'float' solutions are pulled to the same 'fixed' ambiguity vector \( z \). Using the pull-in regions, one can give an explicit expression for the corresponding integer ambiguity estimator. It reads \( \hat{a} = \sum_{z \in \mathbb{Z}^n} z s_z(\hat{a}) \) with the indicator function \( s_z(\hat{a}) \) equal to one if \( \hat{a} \in S_z \) and zero otherwise.

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 is referred to as the class of admissible integer estimators [Teunissen, 1999]. These integer estimators are defined as follows.

Definition

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

\[
(i) \quad \bigcup_{z \in \mathbb{Z}^n} S_z = \mathbb{R}^n \\
(ii) \quad \text{Int}(S_{z_1}) \cap \text{Int}(S_{z_2}) = \emptyset, \forall z_1, z_2 \in \mathbb{Z}^n, z_1 \neq z_2 \\
(iii) \quad S_z = z + S_0, \forall z \in \mathbb{Z}^n
\]

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 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 needed in order to be able to map any 'float' solution \( \hat{a} \in \mathbb{R}^n \) to \( \mathbb{Z}^n \), 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 follows from the requirement that \( S(x + z) = S(x) + z, \forall x \in \mathbb{R}^n, z \in \mathbb{Z}^n \).
Also this condition is a reasonable one to ask for. It states
that when the 'float' solution is perturbed by \( z \in \mathbb{Z}^n \), 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 there-
fore allows one to work with the fractional parts of the en-
tries of \( \hat{a} \), instead of with its complete entries. Important
examples of admissible integer estimators are the estimators
based on the principles of integer rounding, integer
bootstrapping and integer least-squares. In this contribu-
tion we will focus on the principle of integer bootstrapping.

With the division of \( R^n \) into mutually exclusive pull-in
regions, we are in the position to consider the distribution
of \( \hat{a} \). This distribution is of the discrete type and it will be
denoted as \( P(\hat{a} = z) \). It is a probability mass function,
having zero masses at nongrid points and nonzero masses
at some or all grid points. If we denote the continuous
probability density function of \( \hat{a} \) as \( p_a(x) \), the distribution
of \( \hat{a} \) follows as

\[
P(\hat{a} = z) = \int_{S_z} p_a(x) dx , \quad z \in \mathbb{Z}^n
\]

Note that the dependence on the chosen integer estimation
principle enters through the pull-in regions \( S_z \). The above
expression holds for any distribution the 'float' ambiguities
\( \hat{a} \) might have. In most GNSS applications however, one as-
sumes the vector of observables to be normally distributed.
The estimator \( \hat{a} \) is then normally distributed too, with mean
\( a \in \mathbb{Z}^n \) and vc-matrix \( Q_\hat{a} \). Its probability density function
(pdf) reads

\[
p_a(x) = \frac{1}{\sqrt{\det(Q_\hat{a})/(2\pi)^n}} \exp\left\{-\frac{1}{2} \| x - a \|_Q_\hat{a}^2 \right\}
\]

with the squared weighted norm \( \| \cdot \|_Q_\hat{a}^2 = (\cdot)^T Q_\hat{a}^{-1}(\cdot) \).

The probability \( P(\hat{a} = a) \) equals the probability of correct
integer ambiguity estimation, the ambiguity success-
rate. In this contribution we will concentrate on the prin-
ciple of bootstrapping and derive an invariant upperbound
for its success-rate.

2 Integer Bootstrapping

The bootstrapped estimator follows from a sequential con-
tditional least-squares adjustment and it is computed as fol-
loows. If \( n \) ambiguities are available, one starts with the
first ambiguity \( \hat{a}_1 \), and rounds its value to the nearest inte-
ger. Having obtained the integer value of this first ambigu-
ity, 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 correla-
tion with the second ambiguity. This process is continued
until all ambiguities are considered. The components of
the bootstrapped estimator \( \hat{a}_B \) are given as

\[
\hat{a}_{B,1} = [\hat{a}_1] \\
\hat{a}_{B,2} = [\hat{a}_{2|1}] = [\hat{a}_2 - \sigma_21^{-2}(\hat{a}_1 - \hat{a}_{B,1})] \\
\vdots
\hat{a}_{B,n} = [\hat{a}_{n|N}] = [\hat{a}_n - \sum_{j=1}^{n-1} \sigma_{n,j|J}^{-2}(\hat{a}_j|J - \hat{a}_{B,j})]
\]

(4)

where the shorthand notation \( \hat{a}_{i|l} \) stands for the \( i \)th least-
squares ambiguity obtained through a conditioning on the
previous \( I = \{1, \ldots, (i-1)\} \) sequentially rounded am-
biguities, '\( [\cdot] \)' denotes the operation of integer rounding,
\( \sigma_{i,j|J} \) denotes the covariance between \( \hat{a}_i \) and \( \hat{a}_{j|J} \), and \( \sigma_{j|J}^2 \)
denotes variance of \( \hat{a}_{j|J} \). For a review of the theory of in-
teger bootstrapping we refer to [Teunissen, 2001].

The bootstrapped estimator is admissible. The first two
conditions of the definition are satisfied, since - apart from
ties in rounding - any 'float' solution gets mapped to a
unique integer ambiguity vector. Also the third condi-
tion of the definition applies. To see this, let \( \hat{a}^*_B \) be the
bootstrapped estimator which corresponds with \( \hat{a}^* \) = \( \hat{a} - z \). It
follows then from (4) that \( \hat{a}_B = \hat{a}_{B}^* + z \).

The real-valued sequential conditional least-squares so-
lution can be obtained by means of the triangular decompo-
sition of the ambiguity variance-covariance matrix. Let the
triangular decomposition of the variance-covariance ma-
trix be given as \( Q_\hat{a} = LDL^T \), with \( L \) a unit lower trian-
gular matrix and \( D \) a diagonal matrix. Then \( (\hat{a} - z) =
L(\hat{a}^c - z) \), where \( \hat{a}^c \) denotes the conditional least-squares
solution obtained from a sequential conditioning on the en-
tries of \( z \). The variance-covariance matrix of \( \hat{a}^c \) is given
by the diagonal matrix \( D \). This shows, when a compo-
nentwise rounding is applied to \( \hat{a}^c \), that \( z \) is the integer
solution of the bootstrapped method. Thus \( \hat{a}_B \) satisfies
\( L^{-1}(\hat{a} - \hat{a}_B) = 0 \). Hence, if \( c_i \) denotes the \( i \)th canoni-
cal unit vector having a 1 as its \( i \)th entry, the bootstrapped
pull-in regions \( S_{B,z} \) follow as

\[
S_{B,z} = \{ x \in R^n \mid | c_i^T L^{-1}(x - z) | \leq \frac{1}{2}, \quad i = 1, \ldots, n \}, \forall z \in \mathbb{Z}^n
\]

(5)

When using the bootstrapped pull-in region for the proba-
bility mass function (2) with the pdf (3), the success-rate
of integer bootstrapping can be shown to follow as [Teu-
nissen, 1998]

\[
P(\hat{a}_B = a) = \prod_{i=1}^{n} \left( 2\Phi\left( \frac{1}{2\sigma_i|I} \right) - 1 \right)
\]

(6)

where the function \( \Phi \) is defined as \( \Phi(x) =
\int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2} u^2 \right\} du \). This shows that the boot-
strapped success-rate is determined by the conditional
The scalar ambiguity dilution of precision (ADOP) was introduced in [Teunissen, 1997a] as
\[
\text{ADOP} = \sqrt{\det Q_{\tilde{a}}} \quad \text{(cycle)}
\]  
(7)

The ADOP is invariant for the class of admissible ambiguity transformations. An ambiguity transformation \( \tilde{\alpha} = Z\alpha \) is said to be admissible if and only if all the entries of matrix \( Z \) and its inverse are integer. These two conditions are needed in order to retain the integer nature of the ambiguities. It can be shown that the determinant of admissible ambiguity transformations always equals \( \pm 1 \). We therefore have \( \det (Q_{\tilde{\alpha}}) = \det (ZQ_{\alpha}Z^T) \), which shows the invariance of the ADOP. Thus the same ADOP-value is obtained, irrespective of which satellite is chosen as reference in the DD definition of the ambiguities. Likewise, the same ADOP-value is also obtained when one uses, instead of the original variance matrix, the variance matrix of the transformed ambiguities, as produced by the LAMBDA method.

Different approaches can be used for computing the ADOP. First, one may use the variance-covariance matrix of the original DD ambiguities or of any transformed set of ambiguities. Second, for computing the determinant, one may use eigenvalues, conditional variances or, if applicable, the analytical closed form expressions as given in [Teunissen, 1997b].

When the eigenvalues \( \lambda_{\alpha} \) of the ambiguity variance matrix are used, we have
\[
\text{ADOP} = \prod_{i=1}^{n} \lambda_{\alpha}^{\frac{1}{i}}
\]  
(8)

Instead of working with eigenvalues, a cheaper way would be to make use of the conditional variances. This approach is based on using a triangular decomposition or a Cholesky decomposition of the ambiguity variance matrix or its inverse. The entries of the diagonal matrix \( D \) in the \( LDL^T \) decomposition of the variance matrix are the sequential conditional variances of the ambiguities. Since the determinant of the diagonal matrix \( D \) equals the determinant of the variance matrix, the ADOP becomes
\[
\text{ADOP} = \prod_{i=1}^{n} \sigma_{\alpha_i}^{-1}
\]  
(9)

In case of bootstrapping the conditional variances \( \sigma_{\alpha_i}^2 \) are usually already available.

When the ambiguities are completely decorrelated, the ADOP equals the geometric mean of the standard deviations of the ambiguities. This follows from \( \det (Q_{\tilde{\alpha}}) = \prod_{i=1}^{n} \sigma_{\alpha_i}^2 \det (R_{\alpha}) \), where \( R_{\alpha} \) is the ambiguity correlation matrix. Since the decorrelating ambiguity transformation of the LAMBDA method produces ambiguities that are largely decorrelated, the ADOP approximates the average precision of these transformed ambiguities.

### 4 The invariant upperbound and its proof

We now come to the main result of this contribution. As the following theorem shows, the invariant ADOP can be used to obtain an upperbound for the bootstrapped success-rates.

**Theorem (Invariant upperbound)**

For any admissible ambiguity parametrization the bootstrapped success-rate can be bounded from above as
\[
P(\tilde{\alpha}_B = a) = \prod_{i=1}^{n} \left( 2\Phi \left( \frac{1}{2\sigma_{\alpha_i}} \right) - 1 \right) \leq \left( 2\Phi \left( \frac{1}{2\text{ADOP}} \right) - 1 \right)^n
\]  
(10)

where \( \Phi(p) \) denotes the cumulative distribution function of a normal distribution with mean 0 and standard deviation 1.
Note that the upper bound is sharp in the sense that it will be reached when all ambiguities are completely decorrelated. The above easy-to-compute upper bound can be used to decide on the potential usefulness of bootstrapping for ambiguity resolution. If in any application the upper bound turns out to be too small the conclusion must be that one can not expect carrier phase ambiguity resolution to be successful when it is based on the principle of integer bootstrapping.

The above upper bound was introduced without explicit proof in [Teunissen, 1998]. We will now give its proof. The proof is rather lengthy and will therefore be given in a number of steps. For the proof we also make use of two important results which are given in the Appendix.

**Step 1:** The above theorem will be proven by solving the maximization problem

\[
\max_{x_i} \prod_{i=1}^{n} F(x_i) \text{ subject to } \prod_{i=1}^{n} x_i = c \tag{11}
\]

where \( F(x) = 2\Phi(x) - 1, x_i = 1/(2\sigma^2) > 0 \) and \( c \) is a known constant. The constraint has been included to reflect the fact that the product of the ambiguity conditional variances, \( \prod_{i=1}^{n} \sigma^2_i \), is invariant for any arbitrary admissible ambiguity transformation. The constant is therefore given as

\[
c = \left( \frac{1}{2\text{ADOP}} \right)^n \tag{12}
\]

In order to make the above maximization problem more manageable we transform it such that the constraint becomes linear and the objective function can be written as a sum instead of as a product. For that purpose we take the logarithm of the objective function, the logarithm of the constraint and reparametrize by replacing the parameters \( x_i \) with \( x_i = \exp v_i \). As a result we get the maximization problem

\[
\max_{v_i} \sum_{i=1}^{n} \ln F(\exp v_i) \text{ subject to } \sum_{i=1}^{n} v_i = \ln c \tag{13}
\]

We may now think of applying the optimization theorem as given in the Appendix (note: maximizing an objective function is equivalent to minimizing \(-1\) times the objective function). For the theorem to be applicable we need to show that the objective function \( \sum_{i=1}^{n} \ln F(\exp v_i) \) is concave (or \( \sum_{i=1}^{n} - \ln F(\exp v_i) \) is convex) with continuous first derivatives and that the feasible set of the constraint is convex. The latter is easily shown since the constraint is linear. It is also easily shown that the first order derivative of \( F(x) \) is continuous. This leaves us to show that the objective function is concave. But since the sum of concave functions is again concave, it suffices to show that

\[
\ln F(\exp v) \text{ is concave} \tag{14}
\]

**Step 2** We will now use Prekopa’s theorem of the Appendix to show that (14) is indeed true. Since \( F(x) = 2\Phi(x) - 1 = P(u \leq x^2) \), with \( u \) distributed as a central Chi-squared distribution with one degree of freedom, \( u \sim \chi^2(1,0) \), we have \( F(\exp v) = P(\frac{1}{2} \ln u \leq v) \). We therefore need to show that

\[
\ln P\left( \frac{1}{2} \ln u \leq \alpha v + (1 - \alpha)w \right) \geq \alpha \ln P\left( \frac{1}{2} \ln u \leq v \right) + (1 - \alpha) \ln P\left( \frac{1}{2} \ln u \leq w \right) \quad \forall \alpha \in [0,1]
\]

According to Prekopa’s theorem this is true when

\[
\ln p_{\frac{1}{2} \ln u}(\alpha v + (1 - \alpha)w) \geq 
\alpha \ln p_{\frac{1}{2} \ln u}(v) + (1 - \alpha) \ln p_{\frac{1}{2} \ln u}(w) \quad \forall \alpha \in [0,1] \tag{15}
\]

with \( p_{\frac{1}{2} \ln u}(v) \) the pdf of \( \frac{1}{2} \ln u \). In order to verify (15) we need to determine this pdf. Since \( u \sim \chi^2(1,0) \), its pdf is given as \( p_u(x) = (\sqrt{2\Gamma(1/2)^{-1}}x^{-3/2}\exp(-1/2)x, 0 < x < \infty \). Using the transformation \( y = \frac{1}{2} \ln x \) the pdf of \( \frac{1}{2} \ln u \) follows as

\[
p_{\frac{1}{2} \ln u}(y) = \frac{p_u(\exp 2y)}{\frac{1}{2} \exp -2y} = 2 \left( \sqrt{2\Gamma(1/2)} \right)^{-1} \exp y \exp \left\{ -\frac{1}{2} \exp 2y \right\}
\]

We therefore have

\[
\ln p_{\frac{1}{2} \ln u}(y) = \ln \left( 2 \left( \sqrt{2\Gamma(1/2)} \right)^{-1} \right) + y - \frac{1}{2} \exp 2y
\]

which is easily shown to be concave. Therefore (15) is true and by Prekopa’s theorem also (14) is true. The conclusion reads therefore that all conditions for the optimization theorem to be applicable are satisfied.

**Step 3** Application of the optimization theorem as given in the Appendix boils down to finding the solution of \( \partial v_i L(v_i, \lambda) = 0, i = 1, \ldots, n \), with the Lagrangian

\[
L(v_i, \lambda) = \sum_{i=1}^{n} \ln F(\exp v_i) + \lambda \left( \sum_{i=1}^{n} v_i - \ln c \right)
\]

This gives

\[
\partial v_i L(v_i, \lambda) = \frac{F'(\exp v_i)}{F(\exp v_i)} \exp v_i + \lambda = 0, \\
\text{for } i = 1, \ldots, n \tag{16}
\]

in which \( F' \) denotes the first derivative of \( F \). These equations are satisfied when all \( v_i \) are equal. And since
they also need to satisfy the constraint $\sum_{i=1}^{n} v_i = \ln c$, it follows that $v_i = \frac{1}{n} \ln c$, or in terms of the original parameters, that $x_i = c^{1/n}$. Together with (12) this finally gives

$$x_i = \frac{1}{2\text{ADOP}}, \text{ for } i = 1, \ldots, n$$

We have therefore proven that $\prod_{i=1}^{n} F(\frac{1}{2\text{ADOP}})$ is the solution of (11). This concludes the proof of (10).

6 Appendix

In this appendix the two theorems are given which are used in the proof of the invariant upperbound for the bootstrapped success rate.

Theorem (global optimization)
Let $f: R^n \mapsto R$, $f \in C^1$, be a convex function on the set of feasible points

$$\Omega = \{ x \in R^n \mid h(x) = 0 \}$$

where $h: R^n \mapsto R^m$, $h \in C^1$, and $\Omega$ is convex ($C^1$ denotes the class of functions with continuous first order derivatives). Suppose there exists $\hat{x} \in \Omega$ and $\hat{\lambda} \in R^m$ such that

$$\partial_x f(\hat{x}) + \hat{\lambda}^T \partial_x h(\hat{x}) = 0$$

Then $\hat{x}$ is a global minimizer of $f$ over $\Omega$.

Proof: see [Chong and Zak, 1996, p. 379]

Theorem (Prekopa)
Let $p(x)$, $x \in R^n$, be a density function for which

$$\ln p(\alpha x + (1 - \alpha) y) \geq \alpha \ln p(x) + (1 - \alpha) \ln f(y), \forall \alpha \in [0, 1]$$

Then

$$\ln P( x \in A + (1 - \alpha)B ) \geq \alpha \ln P(x \in A) + (1 - \alpha) \ln P(x \in B )$$

where $A, B$ are any sets in $R^n$ and their convex combination is defined as

$$\alpha A + (1 - \alpha) B = \{ z \in R^n \mid z = \alpha x + (1 - \alpha) y, x \in A, y \in B \}$$

and $P$ denotes the probability.

Proof: see [Prekopa, 1971]