The Volume of the GPS Ambiguity Search Space and its Relevance for Integer Ambiguity Resolution

P.J.G. Teunissen, P.J. de Jonge, and C.C.J.M. Tiberius, Geodetic Computing Centre, Delft University of Technology

BIOGRAPHY

Peter Teunissen is professor in Mathematical Geodesy and Positioning. Paul de Jonge and Christian Tiberius1 both graduated at the Faculty of Geodetic Engineering of the Delft University of Technology. They are currently engaged in the development of mathematical models for the GPS data processing in surveying and geodesy.

ABSTRACT

In the theory of integer least-squares ambiguity estimation and validation, a central role is played by the ambiguity search space. In this contribution, the significance of the volume of the ambiguity search space is discussed and analyzed. It is shown how the volume can be used as an estimate for the number of grid points inside the GPS ambiguity search space. This is done for various measurement scenarios, each characterized by the use of different types of data redundancies. For these measurement scenarios, it is also shown how the volume can be used to appropriately scale the size of the ambiguity search space. This analysis is presented in the context of the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method. In particular the low correlation and high precision of the transformed ambiguities as provided by the LAMBDA method, allow an appropriate down scaling of the search space. As a result, only a few candidate grid points are left in the ambiguity search space.

1. INTRODUCTION

In case of short observation time spans, the inclusion into the GPS model of the integer constraints on the carrier phase ambiguities, allows for a dramatic improvement in the precision of the GPS positioning results, provided the estimated integer ambiguities successfully pass the various validation tests. The estimation of the integer ambiguities is not a trivial task however, in particular when computational speed becomes essential, such as in applications in which positioning results are required in near real-time. A vast literature already exists on the problem of efficiently solving for the integer ambiguities. An extensive review of the theory can be found in the book [1] and typical examples of the various methods that have been proposed, can be found in [2-12].

In this contribution we will concentrate on a small, albeit significant, aspect of integer ambiguity estimation: the ambiguity search space and its volume. The ambiguity search space plays a central role in the estimation and validation of the integer ambiguities. It is a multivariate ellipsoidal region, centered at the real-valued least-squares ambiguity vector, from which integer vectors are drawn in the search for the integer least-squares solution. From the point of view of computational efficiency, it is important that the search for the integer least-squares solution, and in case of validation, also the search for the next-best integer solution, can be performed in a speedily manner. We will show that the volume of the ambiguity search space, which itself is very cheap to compute, can be an important aid in this respect.

After a very brief review in section 2 of the theory of least-squares ambiguity decorrelation and estimation, we will show in section 3 for a variety of different measurement scenarios, that the volume of the ambiguity search space can be used as an estimate of the number of grid points that are contained in the search space. We also show, both analytically as well as graphically, how the volume is effected when different types of data redundancy are used.

Since the volume is a good indicator for the number of grid points contained in the search space, it is a useful tool to down size the search space. In this way one can avoid that the search space contains an abundance of unnecessary grid points. We present two approaches for down sizing the search space. The first approach only needs the variance matrix of the ambiguities. Hence, it can already be used in the designing stage prior to the actual measurements. The second approach does depend on the data. It needs the variance matrix and the (real-valued) least-squares solution of the ambiguities. The second approach is superior to the first approach, provided it is used in the context of the LAMBDA method. With the second approach, we
can guarantee that the search space contains at least one grid point, or if needed, at least two grid points, while at the same time the search space itself is likely to remain very small. To illustrate the performance of the second approach, a variety of numerical results, based on different measurement scenarios, are presented and discussed in section 4. As it is shown, the excellent performance is due to the high precision and low correlation of the transformed ambiguities.

2. THE AMBIGUITY SEARCH SPACE

In order for this contribution to be sufficiently self contained, the present section includes a very brief review of the theory of integer ambiguity estimation and the Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method. For more details the reader is referred to the references. For the geodetic community the LAMBDA method was introduced in [13] and for the navigation community in [14]. Details of the computational concepts can be found in [15] and an extensive discussion of the implementation aspects, in [16]. The method is reviewed in [1].

2.1 Integer least-squares estimation

The GPS model on which the estimation of the integer ambiguities is based, reads

\[ y = Aa + Bb + e \]  

(1)

where \( y \) is an \( m \)-vector of ‘observed minus computed’ double-differenced (DD) GPS observables, \( a \) is an \( n \)-vector of unknown integer DD ambiguities, \( b \) is a \( p \)-vector that includes the unknown baseline components, \( e \) is the \( m \)-vector that takes care of the measurement noise and remaining unmodelled effects, and \( A \) and \( B \) are the appropriate design matrices. The observation vector \( y \) may consist of phase-only data, on one or on both of the frequencies \( L_1 \) and \( L_2 \), or it may consist of phase and code data, on \( L_1 \) or on \( L_1 \) and \( L_2 \). The dimension of the unknown parameter vector \( b \) equals \( p = 3 \), in case the model is based on a single stationary baseline without additional unknowns such as ionospheric and/or tropospheric parameters.

In order to obtain estimates for the unknown parameter vectors \( a \) and \( b \), we apply the least-squares principle. The required estimates follow from solving the minimization problem

\[ \min_{a,b} \| y - Aa - Bb \|^2, \quad a \in \mathbb{Z}^n, b \in \mathbb{R}^p \]  

(2)

where \( \| . \|^2 = (.)^T Q^{-1}(.) \), with \( Q \) the variance matrix of the GPS observables. Note that this minimization problem is not an ordinary least-squares problem. Instead it is an integer least-squares problem, due to the inclusion of the integer constraints on the DD ambiguities, \( a \in \mathbb{Z}^n \).

The above integer least-squares problem can be solved in three steps. In the first step, we simply disregard the integer constraint and thus replace \( a \) by \( a \in \mathbb{R}^n \). As a result we obtain an ordinary least-squares problem, which can be solved using standard techniques. The least-squares solution and corresponding variance-covariance matrices that follow from this first step, are denoted as

\[ \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} = \begin{bmatrix} Q_{\hat{a}} & Q_{\hat{a}\hat{b}} \\ Q_{\hat{a}\hat{b}}^T & Q_{\hat{b}} \end{bmatrix} \]  

(3)

This solution is often referred to as the ‘float’ solution.

In the second step, the results \( \hat{a} \) and \( Q_{\hat{a}} \) of the first step are used to compute the integer least-squares estimates of the ambiguities. The integer least-squares estimate of \( a \) will be denoted as \( \bar{a} \) and it is the solution of the minimization problem

\[ \min_{a} (\hat{a} - a)^T Q_{\hat{a}}^{-1}(\hat{a} - a), \quad a \in \mathbb{Z}^n \]  

(4)

Once the minimizer \( \bar{a} \) has been found, the residual \( (\hat{a} - \bar{a}) \) is used to adjust the float solution \( \hat{b} \). This is done in the third step. The adjustment of \( \hat{b} \) provides the least-squares estimate \( \hat{b} \) and corresponding variance matrix as

\[ \begin{align*}
\bar{b} &= \hat{b} - Q_{\hat{b}\bar{a}} Q_{\bar{a}}^{-1}(\hat{a} - \bar{a}) \\
Q_{\bar{b}} &= Q_{\hat{b}} - Q_{\hat{b}\bar{a}} Q_{\bar{a}}^{-1} Q_{\bar{a}\hat{b}}
\end{align*} \]  

(5)

In the error propagation it has been assumed that the integer least-squares ambiguities are non-stochastic. This is, strictly speaking, not true, since also the integer least-squares solution depends on the noise in the data. But if we assume that validation ensures that sufficiently probability mass is located at a single grid point, it follows that the above approximation is permitted for all practical purposes. The least-squares estimates \( \bar{a} \in \mathbb{Z}^n \) and \( \bar{b} \in \mathbb{R}^p \) are often referred to as the ‘fixed’ solution.

The computations needed for the first and third step are rather straightforward and can be based on standard techniques. This is not true however, for the second step. Due to the integer constraint \( a \in \mathbb{Z}^n \), the solution of (4) must be obtained by means of a search. Instead of performing the search in the whole of \( \mathbb{Z}^n \), the search is confined to a local space around \( \bar{a} \). This ambiguity search space is defined as

\[ (\hat{a} - a)^T Q_{\hat{a}}^{-1}(\hat{a} - a) \leq \chi^2 \]  

(6)

It is centered at \( \bar{a} \), its shape and orientation are governed by \( Q_{\bar{a}} \) and its size can be controlled by \( \chi^2 \). In order to be able to guarantee that the search space contains the sought for minimizer \( \bar{a} \), the parameter \( \chi^2 \) should not be chosen too small. But preferably, it should also not be chosen too large. A too large value of \( \chi^2 \) would result in a search space with an abundance of unnecessary grid points. In sections 3 and 4 it will be shown how an appropriate value of \( \chi^2 \) can be chosen.

2.2 The LAMBDA method

Although it is certainly possible to base the search for the integer least-squares ambiguities on the DD search space (6), it is unfortunately a very inefficient approach.
This inefficiency stems from the fact that the DD ambiguities are highly correlated and that the DD search space is very elongated. These drawbacks are absent when the LAMBDA method is applied. This method consists of two main steps.

In the first step the original DD ambiguities are transformed to new ambiguities. The transformed ambiguities have the property that they are less correlated and more precise than the original ambiguities. Let the transformation of the DD ambiguities and their variance matrix be given as

$$z = Z^T a, \quad \tilde{z} = Z^T \hat{a}, \quad Q_z = Z^T Q_a Z$$

(7)

The minimization problem (4) is then equivalent to the minimization problem

$$\min_{\tilde{z}} (\tilde{z} - z)^T Q_z^{-1} (\tilde{z} - z), \quad z \in Z^n$$

(8)

provided matrix $Z$ is integer and volume preserving. With the ambiguity transformation (7), the ambiguity search space transforms accordingly. Thus instead of (6), we now have the transformed search space

$$(\tilde{z} - z)^T Q_z^{-1} (\tilde{z} - z) \leq \chi^2$$

(9)

Since the ambiguity transformation matrix $Z$ is required to be volume preserving, the transformed search space has the same volume as the original DD search space (6). But due to the decorrelating property of the transformation, the new search space is generally far less elongated, and the new variance matrix much closer to a diagonal matrix. In the ideal case of course, we would like the transformed variance matrix to be diagonal. With a diagonal variance matrix, the objective function of (8) would simply become a sum of independent squares in the individual ambiguities. The integer least-squares solution is then obtained from simply rounding the real-valued least-squares estimates to their nearest integer. Complete diagonality of $Q_z$ however, is generally not feasible. This is due to the fact that the entries of $Z$ are required to be integer.

Once the least-squares estimates of the transformed ambiguities and their variance matrix are constructed, the computation of the integer least-squares estimates can commence. This is done in the second step. It is based on a search, using bounds on the individual ambiguities that follow from a sequential conditional least-squares adjustment. These sequential bounds are given as

$$(\tilde{z}_1 - z_1)^2 \leq \sigma_{z_1}^2 \chi^2$$

$$(\tilde{z}_2|1 - z_2)^2 \leq \sigma_{z_2|1}^2 \left( \chi^2 - (\tilde{z}_1 - z_1)^2/\sigma_{z_1}^2 \right)$$

$$\vdots$$

$$(\tilde{z}_{n|N} - z_n)^2 \leq \sigma_{z_{n|b}}^2 \left( \chi^2 - \sum_{j=1}^{n-1} (\tilde{z}_{j|j} - z_j)^2/\sigma_{z_{j|j}}^2 \right)$$

(10)

The notation $\tilde{z}_{j|j}$ is used to denote the least-squares estimate of the ambiguity $z_j$ conditioned on all previous ($j = 1, \ldots, (j - 1)$) ambiguities. Its variance is a conditional variance and it is denoted as $\sigma_{z_{j|j}}^2$.

Note that the above set of $n$ bounds is just an alternative way of describing the ambiguity search space (9). Thus all integer vectors $\tilde{z}$ that satisfy (9) also satisfy (10), and vice versa. The advantage of (10) over (9) is however, that it provides for scalar bounds on the individual ambiguities, thus allowing one to execute the search for the integer least-squares solution.

It will be clear that bounds similar to those of (10) could have been formulated for the original DD ambiguities as well. And with these bounds the search for $\tilde{a}$ can indeed be performed. The big advantage of using the bounds of (10) instead of their counterparts for the DD ambiguities, stems from the signature of the conditional variances of the transformed ambiguities, $\sigma_{z_{j|j}}^2$. It can be shown that DD ambiguities are usually extremely correlated and of a very poor precision. As a result the spectrum of conditional variances of the DD ambiguities will exhibit a large discontinuity, which for a single baseline model with ambiguities and baseline components as only unknowns, can be characterized as

$$\begin{cases} \sigma_{a_{1|j}}^2 \text{ very large for } & i = 1, 2, 3 \\ \sigma_{a_{4|j}}^2 \text{ very small for } & i \geq 4 \end{cases}$$

(11)

This implies, if (10) would have been formulated for the original DD ambiguities, that the first three bounds would be very loose whereas the remaining bounds would be very tight. Hence, the first three bounds would accept many integer candidates, of which however quite some will be rejected again when one arrives at the fourth and following bounds. As a result, search halting is experienced. Search halting is largely eliminated however when one works with the transformed ambiguities. Contrary to the DD ambiguities, the transformed ambiguities are decorrelated to a great extent and are of a very high precision. All conditional variances $\sigma_{a_{i|j}}^2$ are of the same order and very small. Thus instead of only having sharp bounds for the last ($n - 3$) ambiguities, as it is the case with the DD ambiguities, now all $n$ bounds are sharp without any large discontinuity. As a result the search can be performed very efficiently.

As outlined in subsection 2.1, with the integer least-squares solution $\tilde{z}$, the fixed solution is computed: $\hat{b}$ and its variance matrix. Here one can follow two possible routes. Either one computes $\hat{a}$ explicitly from the back transformation $\hat{a} = Z^{-T} \tilde{Z}$ and then uses (5). Or one computes $\hat{b}$ and its variance matrix, without the explicit back transformation, as

$$\begin{cases} \hat{b} = \hat{b} - Q_{h\tilde{z}} Q_{\tilde{z}^{-1}} (\hat{z} - \tilde{z}) \\ Q_{\hat{b}} = Q_{\hat{b}} - Q_{h\tilde{z}} Q_{\tilde{z}^{-1}} Q_{h\tilde{z}} \end{cases}$$

(12)

3. VOLUME OF SEARCH SPACE

In this section we will study some characteristics of the volume of the ambiguity search space. First we will discuss different ways of computing the volume. Then we will consider the volumes of partial search spaces and make a connection with the problem of search halting.
Following this, we will show by means of a variety of numerical examples, that the volume of the ambiguity search space is an excellent indicator for the number of grid points inside the search space. Finally, we show how the volume behaves when different types of GPS data redundancies are used.

3.1 Computing the volume

The volume of the ambiguity search space (6) is given as

\[ V_n = \chi^n U_n \sqrt{\text{det} Q_a} \]  
(13)

where \( n \) is the order of the variance matrix \( Q_a \) and \( U_n \) is the volume of the unit sphere in \( R^n \). The volume of the unit sphere is given as

\[ U_n = \frac{\pi^{n/2}}{\Gamma(n/2 + 1)} \]

where \( \Gamma(x) \) is the gamma-function. The required function values of the gamma-function can be computed using the recurrence relation \( \Gamma(x + 1) = x \Gamma(x) \) and the initial values \( \Gamma(1/2) = \sqrt{\pi} \) and \( \Gamma(1) = 1 \).

For the computation of the determinant of \( Q_a \) one can either use the eigenvalues of the variance matrix or the conditional variances. We have

\[ |Q_a| = \prod_{i=1}^{n} \lambda_{a,i} \prod_{i=1}^{n} \sigma_{a,i}^2 \prod_{i=1}^{n} \sigma_{a,i+1}^2 \]  
(14)

The first product consists of the product of all eigenvalues. The last two are products of conditional variances. In the first product of conditional variances, the conditioning is done on the previous ambiguities, whereas in the second product, the conditioning is done on the ambiguities following. The conditional variances can be obtained from a triangular decomposition of either \( Q_a \) or its inverse. Let a triangular decomposition of \( Q_a^{-1} \) be given as \( Q_a^{-1} = LDL^T \). The conditional variances \( \sigma_{a,i+1,...,n}^2 \) are then given as the entries of the diagonal matrix \( D^{-1} \). The conditional variances \( \sigma_{a,i}^2 \) follow, if instead of a lower triangulation, an upper triangulation of \( Q_a^{-1} \) is used. The lower triangulation can easily be obtained from a Cholesky decomposition. For instance, the Cholesky factor of \( Q_a^{-1} \) is given as \( C_a = L \sqrt{D} \), with \( Q_a^{-1} = C_a C_a^T \).

Since DD ambiguities are defined with respect to a choice of reference satellite, their variance matrix depends on this choice. That is, a different choice of reference satellite, will give a different variance matrix. As a result, also the eigenvalues and the conditional variances are dependent on this choice. The product of the \( n \) eigenvalues and the product of the \( n \) conditional variances however, are invariant to the choice of reference satellite. This is due to the fact that the transformation which changes the choice of reference satellite, is a member of the class of volume preserving transformations. Hence, it leaves the determinant of \( Q_a \) invariant and thus also the products of (14).

Since the decorrelating ambiguity transformation of the LAMBDA method is volume preserving as well, the determinants of the two variance matrices \( Q_a \) and \( Q_\xi \) are identical and thus

\[ |Q_a| = |Q_\xi| = \left\{ \prod_{i=1}^{n} \lambda_{\xi,i} \prod_{i=1}^{n} \sigma_{\xi,i}^2 \prod_{i=1}^{n} \sigma_{\xi,i+1}^2 \right\} \]

(15)

The computation of the volume of the ambiguity search space based on the conditional variances is generally preferable over the computation based on the eigenvalues. Not only are the eigenvalue computations more involved, the triangular decomposition or Cholesky decomposition is usually already available, in particular when solving least-squares problems. The computation of the volume becomes in particular cheap, when the sequential bounds of (10) are constructed. In that case the conditional variances of the transformed ambiguities are readily available.

3.2 Volume of partial search space

For the transformed ambiguities, the partial search space of level \( j \) is defined as

\[ \sum_{i=1}^{j} (z_i - z_i) / \sigma_{z_i}^2 \leq \chi^2 \]  
(16)

Note that this partial search space is described by the first \( j \) bounds of (10). Also note, that the integer minimizers of the left hand side of the above inequality are identical to the integer minimizers of (8), provided that the last \( (n - j) \) ambiguities are relaxed to be reals instead of integers.

We will now consider the volume of the above partial search space as function of the level \( j \). It will be denoted as \( V_j \). Using the volume formula (13) and the fact that the determinant of the variance matrix is given by the product of its conditional variances, it follows that

\[ V_j = \sigma_{z_j} \alpha_j V_{j-1} \]  
(17)

where \( \alpha_j = \sqrt{\pi} \chi \Gamma((j + 1)/2) / \Gamma((j + 2)/2) \). This recursive formula may now be used to compute the volumes of the partial search spaces as the level \( j \) increases from 1 to \( n \). For \( j = n \) it will give the volume of the complete search space.

Similar computations can be done for the volumes of the partial search spaces of the original DD ambiguities. In that case we only need to replace \( \sigma_{z_j} \) in (17) by \( \sigma_{a,i} \). This shows that the differences between the volumes of the two types of partial search spaces are uniquely governed by the signatures of the two types of conditional variances, the ones of the original DD ambiguities and the ones of the transformed ambiguities.

Figure 1 shows a typical example (dual frequency, phase only, 6 satellites) of the volumes of the partial search spaces versus the level \( j \). The full curve shows the volume for the original search space and the dashed curve shows the volume for the transformed search.
It will now also be clear why search halting occurs when the DD ambiguities are used and why it is absent when one uses the decorrelated ambiguities. In the DD case the volume first blows up and then is squeezed down again. Thus first an increasing number of grid points are admitted, and then many of them are rejected again because the volume gets smaller.

### 3.3 Volume and number of grid points

It will be clear that the volume of the ambiguity search space and the number of grid points in it, must be related. The smaller (larger) the volume becomes, the less (more) grid points it generally will admit. It is however far less obvious that the volume itself indicates how many grid points are located within the search space. Of course this can only be an approximation. The volume is namely independent of the location and orientation of the search space, whereas the search space is not. Also, it is not difficult to give synthetic counter examples where the volume is large, while the ellipsoidal region fails to contain a single grid point. But despite these side-notes, one can show analytically that the volume indeed provides on the average a fair approximation of the number of grid points. In this subsection we will show this by means of several numerical examples.

The computations are based on a single baseline model, with a baseline length of about 10 km. Eight different cases are considered. They are shown in table 1. The eight cases differ in the number of satellites used, the types of data used and the number of frequencies used. The capital ‘P’ refers to code measurements, which can be of any type. Figure 2 shows, per case, the actual number of grid points found in the search space versus the volume of the search space. Each case shows the results of ten experiments, each experiment based on an observation time span of one second, using two epochs of data. The time between two experiments within one case is 30 seconds. The standard deviation for the undifferenced phase data was set at 0.3 cm and for the code data at 30 cm. The volumes chosen for the search space, range from 1 to 100, with an increment of one.

When viewing the results for the eight cases, we see a remarkably good agreement between the volume of the search space and the actual number of grid points located in it. In all cases, the lines run under 45 degrees approximately. There are some variations in agreement between the eight cases. For instance in the phase-only case, the single frequency results using 4 satellites show a somewhat better agreement than the dual-frequency results using 6 satellites. This difference is believed to be due to the difference in elongation of the DD ambiguity search spaces. One can show that the first search space is far less elongated than the second search space. This is due to the fact that in the first case, only one frequency is used and no satellite redundancy is available, whereas in the second case the frequencies used are doubled and two more satellites are tracked.

When comparing the phase-only cases with the phase and code cases, we also see that the latter show a somewhat better agreement. Again this is believed to be due to the elongation. When code data are included, the search space is less elongated.

The fact that the volume of the ambiguity search space...
approaches the number of grid points so well, makes it a useful tool, both for setting the size of the ambiguity search space, and for inferring whether one is likely to have a successful validation or not. Note that these computations can be done at the designing stage prior to the actual measurement stage, since no measurements are needed to compute the volume.

In order to set the size of the search space, one proceeds as follows. Depending on the approximate number of grid points required, the volume is set. Then compute the volume of the unit sphere and the determinant of the variance matrix. From $V_{n}$, $U_{n}$ and $|Q_{1}| = |Q_{2}|$, the value of $\chi^{2}$ follows using (13). Of course, since the volume is only an indicator for the number of grid points, it is not so much the precise value that counts, but more its order of magnitude. As to the question of validation, it is not so much the precise value that counts, but more the variance matrix. From computations can be done at the designing stage prior to having a successful validation or not. Note that these volume-curve is shown in figure 3 (top) as function of the number of epochs and $\lambda_{1}$ wavelength. It can exceed as follows. Depending on the approximate number of grids, the variance matrix. From $V_{n}$, $U_{n}$ and $|Q_{1}| = |Q_{2}|$, the value of $\chi^{2}$ follows using (13). Of course, since the volume is only an indicator for the number of grid points, it is not so much the precise value that counts, but more its order of magnitude. As to the question of validation, it is not so much the precise value that counts, but more the variance matrix. From computations can be done at the designing stage prior to having a successful validation or not. Note that these volume-curve is shown in figure 3 (top) as function of the number of epochs and $\lambda_{1}$ wavelength. It can exceed as follows. Depending on the approximate number of grids, the variance matrix.

3.4 Volume and data redundancy

In this subsection we will show how different types of data redundancy effect the volume of the ambiguity search space. The types of data redundancy considered are: satellite redundancy, use of dual frequency data, use of code data and ‘time’. The same eight cases of table 1 are considered. For each of these eight cases, the volume-curve is shown in figure 3 (top) as function of the number of observation epochs and in figure 3 (bottom) as function of the time span in seconds, using two epochs only. The epoch separation is one second. Since it is the qualitative behavior of the volume curves that will be discussed, $\chi^{2}$ has been set at the arbitrary value of one.

We will first discuss the phase-only curves of figure 3 (top). Of the four phase-only curves, the $(L_{1}, m = 4)$ case returns the largest volume; this case also has the least redundancy. There is no satellite redundancy, no code data are used and only a single frequency is used. The volume decreases as time progresses with the number of epochs. This is due to the change in receiver-satellite geometry and to the increase in the number of observation epochs used.

The remaining three phase-only curves can be related to the first curve. Let $V_{m-1}$ denote the volume of the single frequency case using $m$ satellites and let $V_{2(m-1)}$ denote the volume of the dual-frequency case using $m$ satellites. Furthermore let

$$S = \frac{\pi \sigma_{\phi}^{2}}{k \lambda_{1}^{2}}$$  \hspace{1cm} (18)

with $\sigma_{\phi}^{2}$ the variance of the undifferenced phase data, $k$ the number of epochs and $\lambda_{1}$ the $L_{1}$ wavelength. It can then be shown that

$$\frac{V_{m-1}}{V_{2(m-1)}} = \frac{(m-1)(m-2)!}{(m-1)!} \sqrt{\frac{n}{m}} S \left(\frac{m-1}{m+1}\right)$$

$$\frac{V_{2(m-1)}}{V_{2(m-2)}} = \frac{(m-1)(m-2)!}{(m-1)!} \left[\sqrt{\frac{n}{m}} S \left(\frac{m-1}{m+1}\right) \right]^{2}$$  \hspace{1cm} (19)

For the first two equations, $m$ and $n$ have been assumed odd. Similar expressions can be given however for the case they are both even and for the case they are even or odd. The first volume ratio of (19), compares single frequency volumes for different number of satellites $m$ and $n$, the second compares the dual-frequency case with the single frequency case, and the third ratio compares dual-frequency volumes.

Note that the volume ratios only depend on the variables $m$, $n$, $k$ and $\sigma_{\phi}^{2}$, and on the two wavelengths $\lambda_{1}$ and $\lambda_{2}$. This implies that all volume curves can be computed in a rather straightforward manner, once one volume curve is given. It is also very interesting to observe that all three ratios are independent of the receiver-satellite geometry. The volumes themselves of course, are independent that all three ratios are independent.

Finally note that figure 3 (top) also shows that the separation between the volume curves increases. This is due to the increase in the number of observation epochs $k$. And this effect gets amplified when the power of $S$ increases. This shows that the more data redundancy is
available, the more benefit one has from a larger $k$.

We will now discuss the phase and code curves of figure 3 (top). It can be shown that the expressions for the volume ratios given in (19) also hold when code data are included. Hence, these volume ratios are not only independent of the receiver-satellite geometry, but they are also independent of whether code data are used or not. This implies, when using the logarithmic scale for the volume-axis, that all phase and code curves have the same constant offset with respect to their phase-only counterparts. Hence, there is no difference in the relative improvement in volume, when one compares the two types of curves. This is shown in figure 3, by setting the constant offset to zero. The volume itself, of course, does depend on whether code data are used or not. Just like it depends on the receiver-satellite geometry. Hence, the volumes themselves are smaller when code data are included.

Figure 3 also shows that the phase and code curves decrease less rapidly as their phase-only counterparts. This is even more pronounced in figure 3 (bottom), where the volumes are shown as function of the time span. It can be shown that this is due to the fact that the change in receiver-satellite geometry has less impact when code data are included. Thus the inclusion of code data results in a much lower level of the volume curves, but in a less pronounced decrease as time progresses. The amount with which the volume decreases when code data are included, depends on the precision of the code data relative to the precision of the phase data.

4. SCALING THE SEARCH SPACE

In subsection 3.3 we showed how the volume of the ambiguity search space can be used to find a value for $\chi^2$, such that the search space does not contain too many
grid points. This way of choosing $\chi^2$ is independent of the actual data and it can be applied to the DD ambiguity search space as well as to the transformed ambiguity search space. This approach is less suited however, when it is required that the search space must contain a minimal number of grid points. For instance, when the volume is set at, say the value of one, we know that the search space will contain not too many grid points. But we do not know whether it will contain any grid points at all. For such a small value of the volume, the search space may well be empty.

In this section we will present an alternative approach for down sizing the search space. This approach takes the full advantage of the results obtained by the LAMBDA method and it guarantees that only one grid point, or if needed, two grid points are contained in the search space. In contrast to the first approach, this second approach does depend on the actual data. In fact, it works so well since it makes use of the high precision and low correlation of the transformed ambiguities. Therefore the approach performs poorly when it is applied to the original DD ambiguities.

In order to describe the procedure to be followed, we will first discuss the case which guarantees that at least one single grid point is contained in the search space. The idea is simply the following. Starting from the (real-valued) least-squares estimate of the transformed ambiguities, $\hat{z}$, we round each of its $n$ entries to their nearest integer. This will give an integer vector, which then is substituted for $z$ into the quadratic form of (9). The value of $\chi^2$ is then taken to be equal to the value of the quadratic form. This approach guarantees that the search space will at least contain one grid point. Also, the number of grid points contained in it, will not be too large. This is due to the high precision and low correlation of the transformed ambiguities. In fact, it often happens that the so obtained search space only contains one grid point, since in many cases the rounded integer vector already equals the integer least-squares estimate $\hat{z}$. But note that this is not guaranteed, since the variance matrix of $\hat{z}$ is not completely diagonal.

For validation purposes, often not only the most likely integer vector is needed, but also the second most likely integer vector. Hence, in this case we would need to choose $\chi^2$ such that the search space at least contains two grid points and preferably not many more than two. Again the idea of integer-rounding the entries of $\hat{z}$ can be used. As above, we first round all the entries of $\hat{z}$ to their nearest integer. This gives one integer vector. Then another $n$ integer vectors are constructed by rounding all entries of $\hat{z}$ to their nearest integer, except for one entry $\hat{z}_i$, that is rounded to its next-nearest integer ($i$ runs from 1 to $n$). Thus now we have obtained a total of $(n+1)$ integer vectors, all of which have their own corresponding $\chi^2$-value. By setting $\chi^2$ equal to the one-but-smallest of these values, it is guaranteed that the corresponding search space at least contains two grid points, and most likely not many more than two.

In order to show how well the above procedure works for different measurement scenarios, we again consider the eight cases of table 1. And for each of these cases, we used the same data as was used in the experiments of figure 2. For each experiment per case, the $(n+1)$ volumes were computed that correspond to the $(n+1)$ integer vectors. After ordering these $(n+1)$ volumes according to their size, they were plotted in figure 4, for each experiment per case as function of the number of grid points the search space at least contains. Thus for each experiment the smallest volume guarantees one grid point, and the one-but-smallest volume guarantees two grid points, etc.

When viewing the results for the eight cases we observe the appearance of a clear bend in the curves when the data redundancy increases. There is no bend for the $(L_1, m = 4)$ case and the bend is most pronounced for the $(L_1 + L_2 + P_1 + P_2, m = 6)$ case. This can be explained as follows. When the data redundancy increases two things happen. First, the real-valued least-squares solution $\hat{z}$ and the integer least-squares solution $\hat{z}$ will tend to differ less. Also the difference between $\hat{z}$ and the closest rounded integer vector will tend to get smaller. As a result the corresponding volume will get smaller. This explains why the minima of the volume curves tend to get smaller as the data redundancy increases. Secondly, the significance of the difference of the next best integer solution and the real-valued least-squares solution will become larger as data redundancy increases. This explains why the difference between the next-to-smallest volume and the smallest volume gets larger as data redundancy increases.

As to the size of the volume, all eight cases show, with two exceptions, that the volume belonging to the search space that at least contains two grid points, is smaller than 10. The two exceptions occur for the case $(L_1, m = 6)$ and for the case $(L_1 + L_2, m = 4)$. In these two cases, the volume is smaller than 21, which is still very small. These results are important, since they demonstrate that we are able to downside the search space to a very small region, but one that still guarantees to contain at least two grid points.

In fact the results are even better than the plots of figure 4 suggest. In tables 2 and 3 the numerical results are given for the eight cases, including the actual number of grid points that are contained in the search spaces, when $\chi^2$ is set to the one-but-smallest value. Note that in all but one of the experiments, the actual number of grid points located in the down sized search space is very small indeed and of course, by definition, always larger than or equal to two. But even the value of 22 grid points of the ‘outlying’ experiment is small. The two missing values in the last column of each of the two tables, are due to loss of lock.

The above results have been obtained, using the decorrelated least-squares ambiguities $\tilde{z}$. To show that it is indeed a prerequisite that one uses the transformed ambiguities, figure 5 shows a typical result one will get
Figure 4: Volumes of the ambiguity search spaces containing at least $k$ ‘near’ grid points, for the eight cases shown in table 1.

Figure 5: Volumes of the DD ambiguity search spaces containing at least $k$ ‘near’ grid points (case shown: $L_1 + L_2$, $m = 6$).

when the same approach is used, but now applied to the original DD ambiguities. It will be clear from these results, that in this case the procedure has no chance of success at all. That is, when using the original DD ambiguities, the volume values become too extreme to work with. This is due to their poor precision and high correlation.

5. CONCLUDING REMARKS

In this contribution, we studied the volume of the ambiguity search space and its potential in aiding the computational process of estimating the integer carrier phase ambiguities. The volume is invariant to the choice of reference satellite and invariant to the decorrelating ambiguity transformation. Once the conditional variances of the ambiguities are available, only a few computations are needed to obtain the volume of the search space. The conditional variances are readily available from the sequential bounds on which the search for the integer least-squares solution is based.

It was shown for several measurement scenarios, that the volume gives a fair indication of how many grid points are located in the search space. Also the sensitivity of the volume for different types of data redundancy was analyzed. Two approaches were presented for down sizing the ambiguity search space. The first approach can already be used prior to the measurement stage and performs independently from the type of ambiguities used. The second approach depends on the data, but is superior to the first approach, provided it is used within the context of the LAMBDA method. Since it takes the full advantage of the high precision and low correlation of the transformed ambiguities, the ambiguity search space can be scaled down to a very small region, which still guarantees to contain at least one, or if
Table 2: The one-but-smallest volume and the number of grid points (in brackets), for each of the experiments of the phase-only cases shown in figure 4.

<table>
<thead>
<tr>
<th>$L_1$</th>
<th>$L_1 + L_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 4$</td>
<td>$m = 6$</td>
</tr>
<tr>
<td>2.6 (3)</td>
<td>1.3 (2)</td>
</tr>
<tr>
<td>2.2 (2)</td>
<td>4.1 (3)</td>
</tr>
<tr>
<td>1.6 (3)</td>
<td>0.7 (2)</td>
</tr>
<tr>
<td>1.7 (2)</td>
<td>3.8 (2)</td>
</tr>
<tr>
<td>1.4 (2)</td>
<td>1.3 (2)</td>
</tr>
<tr>
<td>1.2 (2)</td>
<td>18.5 (2)</td>
</tr>
<tr>
<td>2.1 (3)</td>
<td>2.6 (3)</td>
</tr>
<tr>
<td>1.4 (2)</td>
<td>0.2 (2)</td>
</tr>
<tr>
<td>1.9 (2)</td>
<td>1.6 (2)</td>
</tr>
<tr>
<td>3.3 (4)</td>
<td>1.2 (2)</td>
</tr>
</tbody>
</table>

Table 3: The one-but-smallest volume and the number of grid points (in brackets), for each of the experiments of the phase and code cases shown in figure 4.

<table>
<thead>
<tr>
<th>$L_1 + P_1$</th>
<th>$L_1 + L_2 + P_1 + P_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 4$</td>
<td>$m = 6$</td>
</tr>
<tr>
<td>2.0 (3)</td>
<td>1.4 (2)</td>
</tr>
<tr>
<td>1.4 (2)</td>
<td>1.6 (2)</td>
</tr>
<tr>
<td>1.6 (2)</td>
<td>4.4 (3)</td>
</tr>
<tr>
<td>1.3 (2)</td>
<td>3.4 (4)</td>
</tr>
<tr>
<td>3.9 (2)</td>
<td>0.4 (2)</td>
</tr>
<tr>
<td>2.1 (2)</td>
<td>1.9 (2)</td>
</tr>
<tr>
<td>1.6 (3)</td>
<td>0.4 (2)</td>
</tr>
<tr>
<td>1.0 (2)</td>
<td>1.4 (3)</td>
</tr>
<tr>
<td>3.1 (3)</td>
<td>2.1 (2)</td>
</tr>
<tr>
<td>4.8 (4)</td>
<td>2.0 (3)</td>
</tr>
</tbody>
</table>

needed, at least two grid points.

In this contribution the LAMBDA method has been applied to a single baseline for which ionospheric and/or tropospheric delays were assumed absent. The method however, is generally applicable. Its use is not restricted to the single baseline case. Without restrictions, it can be applied to network models and GPS models in which additional parameters, other than the baseline components, appear. The method is also applicable to cases where the relative receiver-satellite geometry is dispensed with, the so-called geometry-free models. In each case, the performance of the method is dictated by the signature of the spectrum of the ambiguity conditional variances.

REFERENCES


