Performance of integer parameter estimation algorithm for GPS signals in noisy environment

Mayur Shah, *Arizona State University*
Ying-Cheng Lai, *Arizona State University*

BIOGRAPHY

Mayur Shah received M.S. in Electrical Engineering from Arizona State University in 2004. He received B.E. (Hons.) in Electrical Engineering and M.Sc. (Hons.) Physics from Birla Institute of Technology and Science, Pilani, India, in 2002. He is currently working for Qualcomm.

Ying-Cheng Lai received Ph.D. in Physics from University of Maryland at College Park in 1992. Since 2001 he has been Professor of Electrical Engineering and Professor of Mathematics at Arizona State University. His research areas are nonlinear dynamics, signal processing, complex networks, and computational biology. His honors include a Presidential Early Career Award for Scientists and Engineers (PECASE) in 1997 and election to be a Fellow of the American Physical Society in 1999.

ABSTRACT

Precise positioning utilizing GPS signals relies on a precise tracking of the carrier phase of these signals. The phase generally consists of two parts: an integer part and a fractional part. With the current receiver technology, it is possible to track the fractional part of the phase of the carrier wave at accuracy of 0.001 m using phase locked loops (PLLs). The tracking of the integer part is, however, a more difficult problem because of the inevitable ambiguities associated with the determination of the integer multiple of the carrier wavelength. In an environment where noise is Gaussian with relatively small variance, a standard integer least-squares algorithm can be used, yielding millimeter range accuracy. However, in the presence of interference - intentional or non-intentional, this noise can be large and not necessarily Gaussian. In this paper we evaluate the performance of Integer Least-Squares based algorithm for two different situations (i) Phase measurement noise has larger variance and (ii) noise is not Gaussian.

To study the effect of larger noise variance on the performance of integer least-squares algorithm, we performed the following simulation. For a synthetic GPS setup, Pc is calculated using Monte-Carlo simulations. Values of Pc for various noise variances are plotted versus time. We observed that as noise level increases, the time it takes to achieve a certain value of Pc also increases. This time was observed to vary linearly with the noise amplitude. The significance of this result is that the performance of the integer least-squares algorithm can be predicted for a given noise level. For a desirable performance requirement, our procedure enables the maximum allowable noise power (noise tolerance limit) to be calculated.

We also investigated the performance of the integer least-squares algorithm when the distribution of the measurement noise is not Gaussian. As a particular example uniform noise was considered. To ensure fair comparison between uniform and Gaussian noise, their variances are set to be equal. Simulation results show that the nature of noise has little effect on the performance of the algorithm.

INTRODUCTION

GPS satellites use Direct Sequence Spread Spectrum (DS-SS) signals to transmit information in C/A code and Y-code. These spread spectrum signals have some degree of jamming protection (due to processing gain) built into the signal structure itself. However, since GPS signal is very weak, it is easy for an intentional jammer to overcome the inherent jamming protection of the DS-SS signal. Jamming signals are spread in the frequency domain by the GPS signal de-spreading process. These spectrally spread jammers increase the effective noise floor in a GPS receiver, making difficult precise carrier-phase tracking. Various techniques, both in terms of signal processing (e.g. time-frequency domain processing, subspace processing) and receiver antenna design (e.g. beam forming, null steering) were proposed in literature for jamming protection. However, even with the application of these techniques, typically there are still residual errors in the phase measurement.

The linearized observation equation for double-difference carrier phase measurement can be written as

$$y = A\,x + B\,z + v$$

Where, $y \in R^N$ is the observation, $x \in R^p$ (real) and $z \in Z^q$ (integer) are unknown vectors, known matrices A and B have dimensions N x p and N x q, respectively, $v \in R^N$ is Gaussian N (0, Σ), p is the dimension of position of the receiver, which is usually 3, q is the number of satellites (number of satellites -1 for double differences), N = q x number of epochs, and z is the integer number of wave cycles which needs to be determined. The problem of estimating z is known as integer parameter estimation or integer ambiguity resolution. Once an estimate for z is made, the ambiguity is said to be fixed or resolved, after which the carrier phase observations can be turned into millimeter level range measurements, making it possible to attain high precision positioning solutions.

BACKGROUND

A brief review of the previous work on ambiguity resolution and validation is presented in this section. There are a few publications which deal with literature survey and compare various ambiguity resolution techniques. Some of the review publications include [1] and [2]. Books by Hofmann-Wellenhof [3] and Teunissen [4] have a section devoted to ambiguity resolution and validation techniques. The International Association of Geodesy (IAG) has formed a Special Study Group - SSG 1.157 [5] for GPS ambiguity resolution and validation. SSG 1.157 contains a bibliography list of more than 200 publications. Most of the review papers were published by year 2000. Some new methods have come up since then and some of them are reviewed here.

Comparison of the ambiguity resolution techniques is not easy and not always feasible. The terms of reference of SSG 1.157 clarifies this point:

"Despite the large effort spent by many groups from all over the world in devising various schemes, knowledge about their theoretical foundation, and how the schemes are related to each other, is still lacking. Different terminology is used and comparisons between methods are rare. Due to a lack of knowledge about the various methods, the implementations used in the comparisons (if made at all) are not always complete, thereby making the test results unreliable. Moreover, results reported of one particular method, are often difficult to relate to the results of another method, due to lacking of knowledge of the characteristics of the data and the type of computer that was used."

As a result, comparison of the ambiguity resolution techniques in terms of computational efficiency and performance is not always reliable. The ambiguity resolution techniques can be divided into three categories according to [1].

The first category includes the simplest ambiguity resolution techniques which use C/A-code or P-code pseudoranges directly to determine the ambiguities of the corresponding carrier phase observations. The precision of the code range is not good enough to determine the integer ambiguities and inter frequency linear combinations are used for estimating ambiguities [3].

The second category of algorithms includes the primary ambiguity resolution techniques named Ambiguity Function Method (AFM) [6]. This technique uses only the fractional value of the instantaneous carrier phase measurement and hence the ambiguity function values are not affected by the whole cycle change of the carrier phase or by cycle slips. A brief overview of this method is given in [3].

The third category comprises the most abundant group of techniques which are based on the theory of integer least squares [7]. Parameter estimation under the theory is carried out in three steps - the float solution, the integer ambiguity estimation, and the fixed solution. Each technique makes use of the variance-covariance matrix obtained at the float solution step and employs different ambiguity search processes at the integer ambiguity estimation step. Some of the representative techniques in the category include the Least Squares Ambiguity Search Techniques (LSAST) [8], the Fast Ambiguity Resolution Approach (FARA) [9], the Fast Ambiguity Search Filter (FASF) [10], the Least-squares Ambiguity Decorrelation Adjustment (LAMBDA) [11] and the method by Hassibi and Boyd [12].

At present, LAMBDA is arguably a theoretically sound and practically suitable method among the ambiguity resolution methods. A fairly detailed description of this method is given in [3]. More details on this method can be found in [13]. In summary, the LAMBDA method can be separated into the following steps.

- A conventional least squares is carried out to yield the baseline components (position coordinates) and float ambiguities;
- Using the decorrelating transformation, the ambiguity search space is reparameterized to decorrelate the float ambiguities;
- Using the sequential conditional least-squares adjustment together with a discrete search strategy, the integer ambiguities are estimated. The ambiguities are retransformed to the original ambiguity space;
- The integer ambiguities are fixed as known quantities and another least squares estimate is made for the unknown position coordinates.

The method described in [12] is also based on the integer least squares. The performance of this method is guaranteed with sharp upper and lower bounds. In additions, the complexity of the algorithm is only

polynomial. We have implemented this algorithm for our simulations. The results obtained for the performance in presence of noise are based on the integer least squares principle and they are not dependent on the specific search method used.

A recent paper by Azimi-Sadjadi and Krishnaprasad [14] used Particle Filters (PF) for integer ambiguity resolution. The particle filters are also known as sequential Monte Carlo filters. Usually in particle-filter based approach, the posterior density of unknowns given the observations is approximated by Monte Carlo samples (particles). The basic particle-filter algorithm creates particles from the initial state distribution. As the system evolves according to the state dynamic equations, particles also evolve accordingly. These particles are then weighted according to their proximity to the observations. The particles which agree closely with the observations get higher weights and the others get lower weights. The particle-filter based approach does not require the observation model to be linear. No prior assumptions are made about the PDF of the noise. For details on particle filters, see [15]. The integer ambiguity is treated as a random integer vector to be determined. A method for approximating the conditional probability mass function (pmf) of this integer vector, given the observations, is presented in [14]. The estimate for the integer value is the point that maximizes the pmf.

ESTIMATION PROBLEM

In the linearized observation equation for double-difference carrier phase measurement, y = A x + B z + v, if B = 0, we could have used least-squares method to compute the estimate of x. In order to obtain estimates of both x and z, we consider maximum likelihood (ML) estimation. The ML estimates for x and z are found by maximizing the probability (likelihood) of observing y, i.e.,

$$(x_{ML}, z_{ML}) = \max P_{Y/X,Z}(y/x,z); (x,z) \in R^p \times Z^q.$$

Since v is Gaussian with zero mean and covariance Σ, the probability density of y given x and z is also Gaussian with mean *Ax + Bz* and covariance Σ. The maxima of this quantity can be found by minimizing the following term:

$$(x_{ML}, z_{ML}) = \min(y - Ax - Bz)^T \Sigma^{-1} (y - Ax - Bz).$$

The minimization of this equation gives estimates of x and z, which are called the float solutions as x_{ML} and z_{ML} are real numbers.

If we consider a block matrix [A B] and the unknown vector as $[x\ z]^T$, the float solution is obtained by solving the following equation

$$[A,B]^T \Sigma^{-1} [A,B][\hat{x},\hat{z}]^T = [A,B]^T \Sigma^{-1} y,$$

or,

$$\begin{bmatrix} A^T\Sigma^{-1}A & A^T\Sigma^{-1}B \\ B^T\Sigma^{-1}A & B^T\Sigma^{-1}B \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{z} \end{bmatrix} = \begin{bmatrix} A^T \\ B^T \end{bmatrix} \Sigma^{-1} y.$$

Eliminating x leads to an equation for z,

$$B^T C' B\hat{z} = B^T \Sigma^{-1} y, \quad \text{where}$$
$$C' = \Sigma^{-1}(I - A(A^T\Sigma^{-1}A)^{-1}A^T\Sigma^{-1}.$$

However, our problem is to find the integer *z* to minimize the errors and not the real number vector. This problem of integer minimization can be stated in the following form:

$$z_{ML} = \min(z - \hat{z})^T \Gamma^{-1} (z - \hat{z}); z \in Z^q,$$

where $\Gamma = (B^T C' B)^{-1}$

Clearly, the minimum over real vector set is zero, when z is equal to $\hat{z}$. We are looking for an integer vector z, which is closes to $\hat{z}$. A simple approach is to round each component of $\hat{z}$ to its nearest integer. This simple approach works when Γ is diagonal. However, in practice, Γ is not diagonal and ambiguities are coupled. As a result, simple rounding off does not yield optimal results.

Once the minima over $z \in Z^q$ is found, maximum likelihood estimate of x is found by substituting the value for z into the least-squares equation. We obtain

$$x_{ML} = (A^T\Sigma^{-1}A)^{-1}A^T\Sigma^{-1}(y - Bz)$$

It is clear that is the covariance matrix for $\hat{z}$ and z is its mean. Since, $\hat{z}$ is a linear function of Gaussian variable y, it is Gaussian too. We write $\hat{z} = z + u$ where *u* is Gaussian with zero mean and covariance matrix Γ. Multiplying both sides by $G = \Gamma^{-1/2}$ and by defining , we get $\hat{y} = Gz + \hat{u}$ where $\hat{u}$ is a Gaussian random variable with zero mean and unit variance. This equation can be written in an equivalent form

$$z_{ML} = \min \| \hat{y} - Gz \|^2; z \in Z^q.$$

The set $\{Gz \mid z \in Z^q\}$ is a lattice in R^q. This equation suggests that the maximum likelihood value of z is found by computing the nearest lattice point to the vector $\hat{y}$.

LATTICES AND BASIS REDUCTION

In this section, we explain the basics of lattice theory required to understand its applications to integer ambiguity resolution problem in GPS. In lattice theory, a

generator matrix G is any matrix with real entries whose rows are linearly independent. The lattice generated by G is

$$L(G) = \{Gz \mid z \in Z^q\}$$

The rows of G are called basis vectors for L, and the number of basis vectors q is said to be the dimension of the lattice L.

The *closest point problem* is the problem of finding, for a lattice L and a given input point $x \in R^q$, a vector $\hat{c} \in L$ such that

$$\| x - \hat{c} \| \leq \| x - c \|, \forall c \in L,$$

where || || represents a Euclidean norm. This problem is known to be NP-hard [16] (i.e. no polynomial time algorithm is known to exist to solve this problem).

The *Voronoi region* or a *Voronoi cell* of a lattice point is the set of all points closer to lattice point c than any other point in the lattice. Mathematically, the region can be expressed as

$$\Omega(L,c) = \{\| x - \hat{c} \| \leq \| x - c \|, \forall c' \in L \mid x \in R^q\},$$

where $c \in L$. The *Voronoi diagram* of a lattice is the set of all its Voronoi regions. It is known that Voronoi regions are symmetrical with respect to reflection in c, and that they are translations of (L,0), where 0 is the origin of R^q.

The *shortest lattice vector* or the *minimum distance* of the lattice d_{min} is defined as

$$d_{\min} = \min \| Gz \|; z \in Z^q.$$

To understand these definitions, consider the following lattice generator matrix as an example:

$$G_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The rows of G_1, [1 0] and [0 1] are the basis vectors as shown in Fig. 1. The Voronoi cells for the same lattice are shown as the thick square blocks in Fig. 2. All Voronoi cells are translations of the cell at origin, which is shown by the shaded region. Note that the basis vectors for this lattice are orthogonal to each other and for a given floating point; the nearest lattice point can be found by simply rounding it to the nearest integer. Two lattices are said to be identical if all lattice points are the same.

Two generator matrices G_1 and G_2 generate identical lattices $L(G_1)$ and $L(G_2)$ if and only if

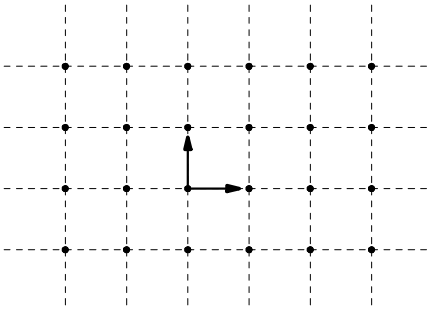

Figure 1. Basis vectors for a square lattice.

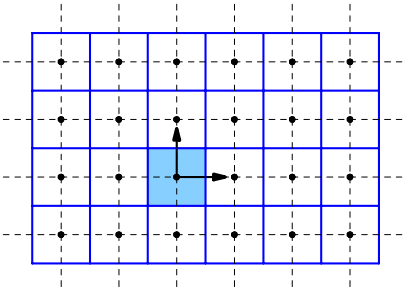

Figure 2. Voronoi cell for a square lattice.

$$G_2 = FG_1,$$

where F is a square matrix with integer entries such that

$$| \det F | = 1.$$

Now consider another generator matrix

$$G_2 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

This matrix can be written in the form of

$$G_2 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = FG_1$$

Since F contains integer entries and | det F | = 1, lattices generated by G_1 and G_2 are identical. The lattice generated by G_2 with its basis vectors and Voronoi regions is shown in Fig. 3. Note that basis vectors for generator matrix G_2 are not orthogonal to each other. Hence finding the nearest lattice point problem is no longer a simple task. Consider a floating point in space as shown in Fig. 4. The nearest lattice point is point 1, but rounding to the nearest integer will give point 2, which is not the correct solution.

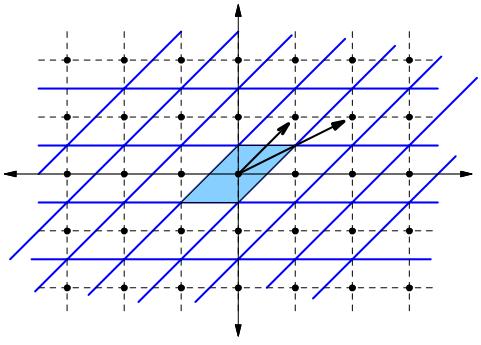

Figure 3. Lattice generated by generator matrix G_2

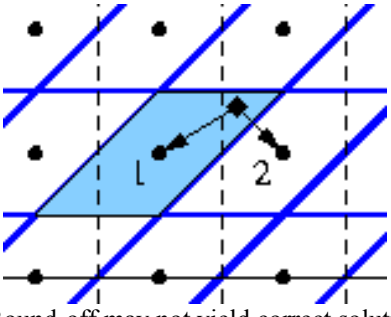

Figure 4. Round-off may not yield correct solution when lattice is not orthogonal.

It is evident from this example that the nearest lattice point can be obtained easily only if the basis vectors are orthogonal.

The process of selecting a good basis for a given lattice, given some criterion, is called *reduction*. In our application, it is advantageous if the basis vectors are as short as possible and "reasonably" orthogonal to each other. Two methods for reduction that are widely used in practice are the Korkine-Zolotare (KZ) and the Lenstra-Lenstra-Lovasz (LLL) method [17]. One reason for their popularity is that with both the mentioned criteria, the n-dimensional reduction problem can be recursively reduced to (n-1)-dimensional reduction problem. Any KZ reduced generator matrix is also LLL reduced. LLL reduction is often used in situations where KZ reduction would be too time-consuming. A comparison of average search times among different reduction methods is made in [17]. It is shown that, up to about 10-15 dimensions, the polynomial-time LLL reduction gives good results. In GPS related problems, the dimension of generator matrix is smaller than this. Hence, LLL algorithm is preferred.

VERIFICATION PROBLEM

Once the estimates of the unknown vectors are made, it is equally important to verify them. Since z is an integer valued vector, we can estimate it exactly. Hence we can define the probability to estimate z correctly as

$$P_c = \text{Probability}(z_{ML} = z).$$

Now, x is a real valued vector. We can compute the probability of our estimate lying in a Euclidean ball of small radius of its actual value, i.e.,

$$\text{Probability}(\| x - x_{ML} \| < \varepsilon).$$

Recall equation

$$\hat{y} = Gz + \hat{u}$$

where $\hat{u}$ is Gaussian random variable with zero mean and unit variance and the set $\{Gz \mid z \in Z^q\}$ is a lattice in R^q. It is evident from this equation that $\hat{y}$ is off from G z by $\hat{u}$. So if $\hat{u}$ is small enough that $\hat{y}$ remains in the Voronoi cell of lattice point G z, the estimate of z is correct. Because of the periodic structure of lattice, the Voronoi cell of lattice point G z is the translation of the Voronoi cell at the origin of the lattice by vector G z. If we call the Voronoi cell at the origin of the lattice V_0, then we can write

$$P_c = \text{Probability}(\hat{u} \in V_0)$$

Hence P_c is equal to the probability of a q-dimensional normal random variable falling inside the Voronoi cell V_0.

The calculation of Pc requires integration of a normal probability density function over the Voronoi cell. The problem of finding the Voronoi cell of the lattice and performing the integration is a very computationally challenging task. Therefore, even if it is possible to calculate P_c exactly, it may not be feasible in many applications. However, sharp upper and lower bounds on P_c can be found with very low complexity. An upper bound on P_c can be found using the modulus value of the determinant of G (|det G|) [12]. The volume of the Voronoi cell is given by |det G|. The larger the value of |det G|, the larger the probability of correct integer estimation. The shape of the Voronoi cell is also a factor determining P_c. Among all shapes with a given volume, the one that maximizes the probability (under normal distribution) is a Euclidean ball. Therefore, the probability of a Euclidean ball of volume equal to |det G| gives an upper bound on Pc. The radius of a q-dimensional Euclidean ball of volume |det G| is equal to $(|\det G|/q)^{1/q}$, and the upper bound on Pc becomes

$$P_{c,up} = \text{Probability}(\| w \| < (|\det G|/\alpha^q)^{1/q}),$$

Where $w \sim N(0, I)$ and $\alpha_q = \pi^{q/2} / \Gamma(q/2+1)$. The sum of squares of q independent zero-mean, unit variance normally distributed random variables is a χ^2 distribution with q degrees of freedom [7]. Hence, we get,

$$P_{c,up} = F_{\chi^2}((|\det G|/\alpha^q)^{2/q}; q).$$

A lower bound on Pc can be found using the shortest lattice vector or the minimum distance d_{min} of the lattice, where d_{min} is actually the distance between the origin and its closest neighbor in the lattice. A ball of radius $d_{min}/2$ is

the largest bass centered at the origin that lies in V_0. The probability of noise falling in a ball centered at the origin of lattice with radius $d_{min}/2$, gives us a lower bound on Pc. Computing d_{min} is an NP-hard problem. A lower bound on d_{min} can be found by using the Gram-Schmidt orthogonalization method [12]. Hence the lower bound on Pc is given by

$$P_{c,low} = \text{Probability}(\| w \| < d/2),$$

where w is N(0; I) and $d<d_{min}$. In terms of χ^2 cumulative distribution function, we have

$$P_{c,low} = F_{\chi^2}(d^2/4; q).$$

SIMULATOIN RESULTS

The simulations in this section are based on a synthetic 2-D GPS setup (Fig. 5). This setup is very similar to the one used in [12]. We assume that the position of the (GPS) receiver x to be determined can be modeled as a zero-mean Gaussian random variable with variance σ_x^2 in each dimension. The coordinate axes are chosen such that the origin is a point on the surface of the earth (a point on the periphery of a circle of radius equal to that of the earth Re = 6357 km). We suppose that there are three visible satellites orbiting the earth with an altitude of 20 200 km and with a period of 12 h (angular velocity of 1/120 s^{-1}). The satellites transmit a carrier signal of wavelength = 19 cm each, and their coordinates are known to the receiver. The receiver, which is assumed to be completely synchronized with the satellites (meaning that it can generate the transmitted carrier signals), measures the phase of the received carrier signals every T = 2 s and unwraps them as times goes by. By multiplying these (unwrapped) phase measurements by the wavelength divided by 2π, the receiver can measure its distance (or range) to each satellite up to some additive noise, which is assumed to be $N(0, \sigma^2)$ and, of course, up to an integer multiple of the wavelength. (This integer multiple can be thought as the number of carrier signal cycles between the receiver and the satellite when the carrier signal is initially phase locked.) By linearizing the range equations, the problem becomes one of estimating a real parameter x (the coordinates of the receiver) and an integer parameter z (the integer multiples of the wavelengths) in a linear model. In the simulation that follows, the actual location of the receiver is $x = [50; 100]^T$, which will be estimated using the carrier phase measurements. We assume that the standard deviation of x is σ_x = 100 m along each coordinate axis.

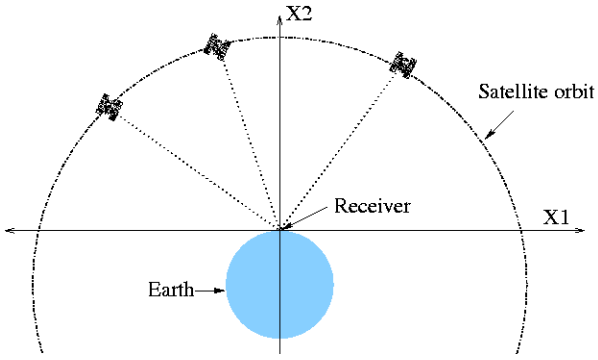

Figure 5. Synthetic 2-D GPS setup

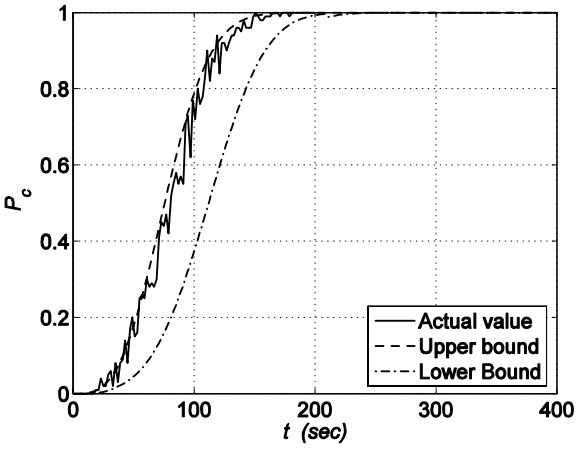

Figure 6. P_c versus time with upper and lower bound for σ=0.01m.

The satellites make angles of 100, 130, and 50 degrees with the axis initially, and the direction of rotation for all of them is clockwise. The variance of phase measurement noise in units of length is taken as 0.01 m. Using the carrier phase measurements taken over a period of 400 s, the receiver tries to find its position x (as well as the ambiguous integer multiples of the wavelengths) as a function of time by solving for the ML estimates. Figure 6 shows the performance of the integer least-squares algorithm for σ = 0.01 m. The exact value of the P_c (represented by solid curve) is computed by Monte Carlo simulations using 1500 random variables.

The upper bound on P_c is computed using |det G|, as represented by the dashed line. It is evident that the upper bound is very sharp. The lower bound on Pc is computed using the lower bound on d_{min}, as shown by a dash-dot curve in Fig. 6. We see that the lower bound is not sharp to begin with, but as Pc approaches unity, it gets sharper. Hence, the lower bound is sharper where it matters the most.

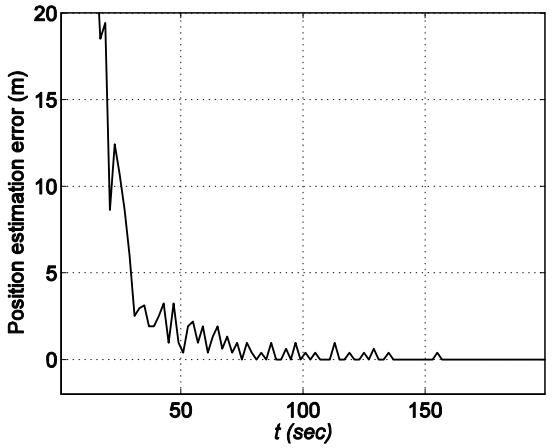

Figure 7. Position estimation error (meters) versus time (sec).

Figure 7 shows the results of the algorithm in terms of receiver positioning. The estimation error is plotted as a function of time. It can be seen from Figs. 6 and 7 that the position estimation error reduces as the probability P_c of correct integer estimation approaches unity. When the integer ambiguity is resolved correctly, the position estimation error is of the order of millimeters.

Having established the accuracy of the integer least squares algorithm for static GPS positioning, we with to evaluate its performance in the presence of noise. For this purpose, the algorithm was provided with various values of phase measurement noise variance as input.

The range of σ is varied from 0.002 m to 0.02 m with increment of 0.002 m. The values of P_c for all cases were calculated using 1500 Monte Carlo simulations. Figure 8 shows the curves of P_c plotted against time for various σ values. It is observed that as σ increases, the time it takes to achieve a certain Pc also increases. The family of curves in Fig. 8 can be used to calculate the maximum allowable noise variance for a given amount of observation time and required value of P_c. For example in Fig. 8, if the observation time is 150 s and the integer ambiguity estimate is to be reliable with 90 per cent accuracy (P_c=0.9), the maximum allowable noise variance is about 0.012 m.

A different approach can be used to analyze the data. Let T_{Pc} be the observation time required to achieve the probability of correct ambiguity resolution P_c. Figure 9 shows T_{Pc} versus σ for various P_c values (P_c=0.5, 0.75,0.90,0.95,0.99).

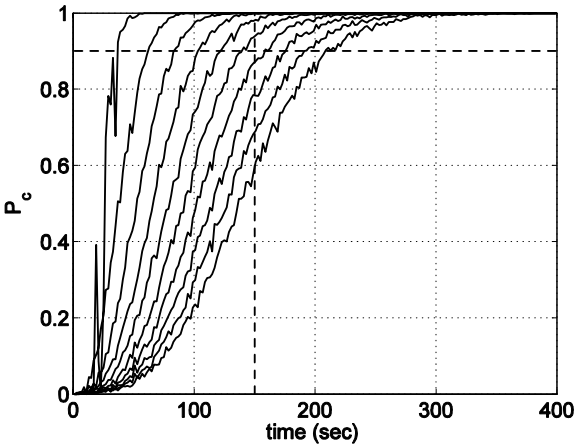

Figure 8. P_c versus time for σ values ranging from 0.002 m to 0.02 m. Maximum allowable noise variance for given values of time and P_c can be determined by the dashed lines.

The 'dots' in the plot represent the actual data points, which can be represented by linear fits. That is we have

$$T_{Pc} \propto \sigma$$

This linear relation characterizes the sensitivity of performance to measurement noise variance. The significance of this results is that the performance of integer least-squares algorithm can be predicted of a given noise level. The plots for |det G| versus σ at three different time instances t = 50 s, t = 100 s and t = 150 s are shown in Fig. 10. It is observed from these plots that as σ increases, |det G| decreases. Theoretically, the effect of noise on P_c can be assessed, as follows. The upper bound on P_c can be calculated using the χ^2 cumulative distribution function (cdf), whose argument is a linear function of the determinant of G. Since χ^2 cdf is a non-decreasing function, a smaller value of |det G| will result in lower P_c. Therefore, as the noise variance increases, |det G| decreases, so does P_c. Since |det G| is the volume of the Voronoi cell, increasing the noise variance is equivalent to shrinking the Voronoi cells. As a result, under noise the lattice points are closer to each other, thereby increasing the probability of error.

We wish to evaluate the effect of number of satellites on the performance of integer least squares algorithm. Since the upper bound on P_c is very sharp, it is used instead of evaluating the exact values of P_c using Monte Carlo simulations. Figure 11 shows the simulation result for upper bound on P_c versus time for 3, 4, and 5 satellites.

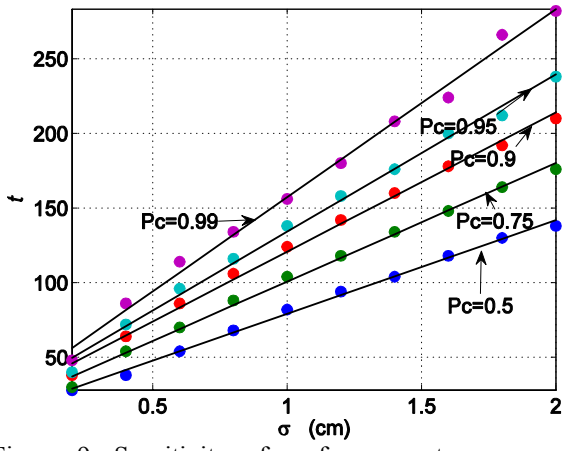

Figure 9. Sensitivity of performance to measurement noise variance.

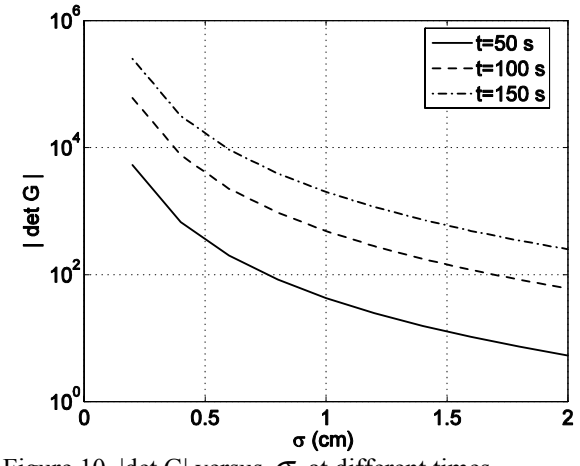

Figure 10. |det G| versus σ at different times

It is seen from the plots that increase in the number of satellites improves the performance of algorithm.

So far in all our discussions, we have assumed the probability density function of the noise to be Gaussian. It may happen in reality that the measurement noise does not follow Gaussian distribution. As a particular example uniform noise is considered. To ensure fair comparison between them, their powers are kept equal. We assume both the noise distributions to be zero mean. If the Gaussian distribution has zero mean and variance σ^2, and if have uniform noise density in the interval [-k; k] with amplitude 1/2k, the variance of this probability density function is $k^2/3$. We thus have $k \approx 1.73\sigma$.

Figure 12 shows the performance comparison for both noise densities for two different noise variances (σ = 0.01 m and σ = 0.02 m).

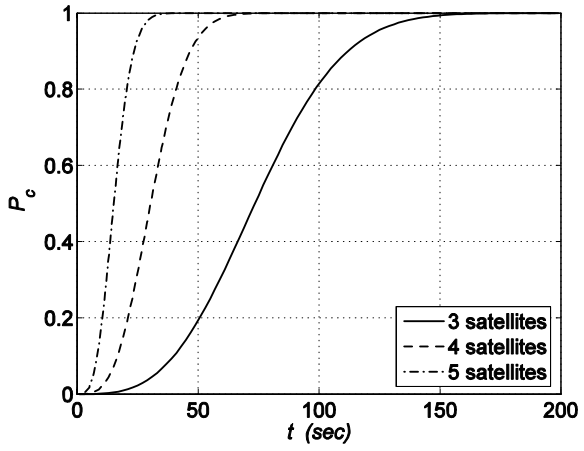

Figure 11. P_c (upper bound) versus time for various satellites

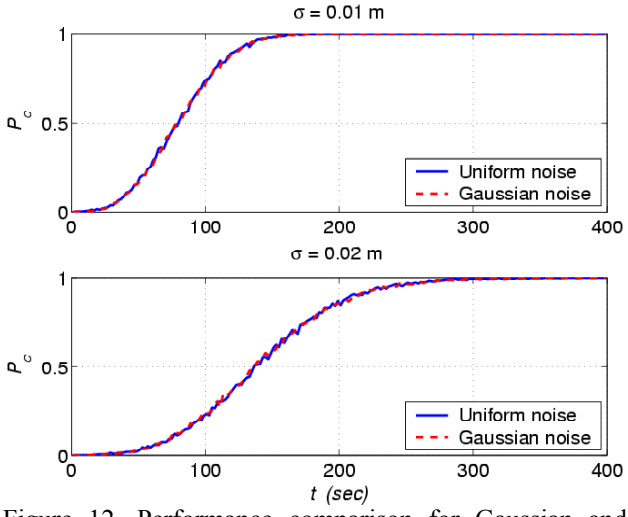

Figure 12. Performance comparison for Gaussian and Uniform noise

The performance is identical for both uniform and Gaussian noise densities for both cases. A possible explanation for this behavior is as follows. If we used the regular least squares techniques to solve for unknowns in linear model and Gaussian noise, the optimal estimator can be easily found.

For the case where noise is Uniformly distributed, this estimator may not be the optimal but it is still the best linear unbiased estimator. Further, the integer least squares method involves a grid search of integer points around the suboptimal solution. Therefore, it is reasonable that we have found the optimal solution for both uniform and Gaussian noise cases.

CONCLUSIONS

This paper addresses the performance of integer least-squares algorithm for GPS signals in noisy environments. It is shown that increasing the noise variance results in shrinking the Voronoi cells. Hence, under noise the lattice points are closer to each other, thereby increasing the probability of error. We find that the observation time required to achieve a fixed value of P_c is directly proportional to standard deviation of phase measurement noise. From the simulations, we can also conclude that the nature of noise has little effect on the performance of the algorithm.

For future directions, the study for the effect on noise on GPS positioning should be extended to kinematic GPS positioning algorithms. It is also recommended to develop a particle-filter based algorithm and compare its sensitivity to noise to that of integer least-squares algorithm for both static and kinematic positioning.

ACKNOWLEDGEMENTS

This work was supported by AFOSR under Grant No. FA9550-04-1-0115.

REFERENCES

[1] D. Kim, and R. Langley, "GPS Ambiguity Resolution and Validation: Methodologies, Trends and Issues", International Symposium on GPS/GNSS, Seol, Korea, Nov. 30 - Dec. 2, 2000.

[2] R. Hatch, and H.-J. Euler, "Comparison of several AROF kinematic techniques", Proceedings of ION GPS-94, Salt Lake City, Utah, September 1994, pp. 363-370.

[3] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, Global Positioning System, Theory and Practice, 4th ed. New York: Springer- Verlag, 1997.

[4] A. Kleusberg and P. J. G. Teunissen, GPS for Geodesy, Lecture Notes in Earth Sciences. New York: Springer-Verlag, 1996.

[5] IAG SSG 1.157, "GPS ambiguity resolution and validation. Preliminary Report",

[6] C. C. Counselman, and S. A. Gourevitch, "Miniature interferometer terminals for earth surveying: ambiguity and multipath with Global Positioning System", IEEE Transactions on Geo-science and Remote Sensing, Vol. GE-19, No. 4, 1981, pp. 244-252.

[7] P. J. G. Teunissen, Least-Squares Estimation of the Integer GPS Ambiguities, LGR Series, Delft Geodetic Computer Centre, 1993.

[8] R. Hatch, "Instantaneous ambiguity resolution", Proceedings of KIS 90, Ban_, Canada, 10-13 September, pp. 299-308,1990.

[9] E. Frei, and G. Beutler, "Rapid static positioning based on the fast ambiguity resolution approach "FARA": theory and first results", Manuscripta Geodaetica, Vol. 15, No. 4, 1990, pp. 325-356.

[10] D. Chen, and G. Lachapelle, "A comparison of the FASF and least-squares search algorithms for on-the-y ambiguity resolution", Navigation: Journal of The Institute of Navigation, 1995, Vol. 42, No. 2, pp. 371-390.

[11] P.J.G. Teunissen, "A new method for fast carrier phase ambiguity estimation", Proc. IEEE Position, Location and Navigation Symp., Las Vegas, NV, 1994.

[12] A. Hassibi and S. Boyd, "Integer Parameter Estimation in Linear Models with Applications to GPS", IEEE Trans. on Signal Processing 46(11):2938-52, Nov. 1998.

[13] P. J. de Jonge and C. C. J. M. Tiberius, The LAMBDA Method for Integer Ambiguity Estimation: Implementation Aspects, vol. 12 of LGR-Series, Delft Geodetic Computing Centre, 1996.

[14] B. Azimi-Sadjadi and P.S. Krishnaprasad, "Integer ambiguity resolution in GPS using particle filtering", Proceedings of the American Control Conference, Volume: 5 , 25-27 June 2001 vol.5 pp:3761-66

[15] M.S. Arulampalam et.al., "A tutorial on particle filters for online nonlinear/non- Gaussian Bayesian tracking" IEEE Transactions on Signal Processing, Vol 50, Feb. 2002 pp 174 - 188

[16] P. van Emde Boas, "Another NP-complete partition and the complexity of computing short vectors in a lattice", Tech. Rep. 81-04, Math. Inst., Univ. Amsterdam, Amsterdam, The Netherlands, 1981.

[17] E. Agrell et. al. "Closest Point Search in Lattices", IEEE Trans. on Information Theory vol. 48(8), pp 2201-2214, Aug. 2002.