A Recursive Least Squares Approach for Carrier Phase Based Positioning

Xiao-Wen Chang, Christopher C. Paige, and Liang Qiu, *McGill University, Canada*

BIOGRAPHIES

Dr. Chang is an Assistant Professor of Computer Science at McGill University. He holds a B.Sc and an M.Sc in Computational Mathematics from Nanjing University, China, and a Ph.D in Computer Science from McGill University. His main research area is scientific computing with particular emphasis on matrix computations and applications. He has been involved in GPS research since 1998.

Dr. Paige is Professor Emeritus at McGill University, School of Computer Science. He holds a B.Sc and a B.Eng. (Electrical) from the University of Sydney, Australia, and a Ph.D in Computer Science from the University of London. His main research area is numerical linear algebra and its applications, and one of his interests is efficient and numerically reliable estimation for linear systems.

Mr. Qiu is an M.Sc student in the School of Computer Science, McGill University. He has a B.Sc in Computer Science from McGill University.

ABSTRACT

In this paper a recursive least squares technique is used for carrier phase based GPS positioning. It is based on single differenced carrier phase measurement equations. We take full account of the structure of the problem to make our algorithm efficient, and use orthogonal transformation techniques to ensure that the computation is numerically reliable. Our algorithm usually converges slightly faster than the van Graas and Lee algorithm [12], which uses the so-called complementary Kalman filter technique. A significant difference between the two algorithms is that ours is more amenable to analysis, and requires less computation in each epoch. We also think it is simpler to understand and implement.

We argue that such numerically stable implementations of the recursive least squares approach applied to single differenced carrier phase measurements provide the most effective tools for this class of problems.

1 INTRODUCTION

Many people have studied using only carrier phase measurements for dynamic position estimation, see for example [1], [3], [4], [6], [7], [12] *et al.*. In practical usage, superior results can probably be obtained by combining both code and carrier phase measurements, but it is important to find the best way of handling each separately, and here we will consider carrier phase measurements only. In [12] van Graas and Lee present the "complementary Kalman filter" approach for obtaining estimates from carrier phase measurements. In their approach, double differenced carrier phase measurements are used to construct measurement equations, and triple differenced measurements are used to propagate the user position, which, according to [12] is the key to achieving sub-meter accuracies with an initialization time of less than 1 minute. Essentially the triple differences implicitly provide the state equations which are needed in Kalman filtering. However the state equations do not provide any new information, since they are constructed from the double differenced measurement equations. In our view the Kalman filtering technique is not necessary in this case, unless perhaps additional information on the velocity of the user is available. Artificially constructing a Kalman filter may delay the convergence, and the additional computation in each epoch is unnecessarily expensive. In this paper, which is based on single difference carrier phase measurement equations, we use least squares (LS) rather than Kalman filtering to estimate the position. In designing the algorithm we make full use of the structure to make the algorithm efficient. Orthogonal transformation techniques are used to ensure that

the algorithm is numerically reliable. Our simulations suggest our approach usually converges slightly faster than the van Graas and Lee approach. Also in each epoch the computational cost of the former is about half that of the latter.

This paper is organized as follows. In Section 2, for completeness we derive the mathematic model for position estimation. In Section 3 we present an efficient and numerically reliable algorithm for computing the position estimate. In Section 4 some simulation test results are given, together with comments on preliminary real data computations. Finally some remarks and conclusions are given in Section 5.

Notation used. We work with reals only, and use i, j, k, l, m, n to denote indices and dimensions (superscript i will refer to the i-th satellite, subscript k to the k-th epoch) while lower case Greek letters will denote scalars. Other lower case Roman letters will denote vectors, while upper case Roman will denote matrices. One exception is that for the integer ambiguities we use N to follow the tradition in the GPS literature. Superscript ^{T} will denote transpose, and $A^\dagger$ is the Moore-Penrose generalized inverse of A. The unit matrix will be denoted by I and its i-th column by e_i, while $e \equiv (1, 1, \ldots, 1)^T$ (we use $\equiv$ to mean 'is defined to be'). I_n will denote the $n \times n$ unit matrix. Throughout we use the norm $\|x\| = \|x\|_2 = \sqrt{x^T x}$ for vectors. We will use $\mathcal{E}\{\cdot\}$ to denote the expected value, and $\mathrm{cov}\{\cdot\}$ to denote the covariance, that is $\mathrm{cov}\{x\} = \mathcal{E}\{(x - \mathcal{E}\{x\})(x - \mathcal{E}\{x\})^T\}$. $v \sim \mathcal{N}(\bar{v}, V)$ will mean v is a a normally distributed random vector with mean $\bar{v}$ and covariance V.

2 THE MATHEMATICAL MODEL

Suppose the baseline (the distance between the stationary receiver and the roving receiver) is short. Then single differencing (between the two receivers) will make the atmospheric refraction, the satellite clock errors, and the satellite ephemeris errors negligible. At epoch k, the single difference carrier phase measurement with respect to satellite i satisfies

$$\phi_k^i = \lambda^{-1}(e_k^i)^T x_k + N^i + \beta_k + \nu_k^i. \tag{1}$$

where

ϕ_k^i is the single difference carrier phase measurement in wavelength;

λ is the wave length of the carrier L1 in meters;

x_k is the baseline vector in meters pointing from the stationary receiver to the roving receiver.

e_k^i is the unit vector pointing from the middle of the baseline vector to the i-th satellite;

N^i is the unknown single difference integer ambiguity;

β_k is the difference between the two receivers' clock biases in wavelength;

ν_k^i is the single difference noise.

In the usual model, the ν_k^i for different satellites and different epochs are assumed to be independently normally distributed. Suppose there are m visible satellites at epoch k, then writing

$$y_k \equiv \begin{bmatrix} \phi_k^1 \\ \cdot \\ \phi_k^m \end{bmatrix}, \quad E_k \equiv \lambda^{-1} \begin{bmatrix} (e_k^1)^T \\ \cdot \\ (e_k^m)^T \end{bmatrix},$$
$$N \equiv \begin{bmatrix} N^1 \\ \cdot \\ N^m \end{bmatrix}, \quad v_k \equiv \begin{bmatrix} \nu_k^1 \\ \cdot \\ \nu_k^m \end{bmatrix},$$

we have

$$\begin{aligned} y_k &= E_k\, x_k + N + e\beta_k + v_k, \\ v_k &\sim \mathcal{N}(0, \sigma^2\, I_m). \end{aligned} \tag{2}$$

This is the desired single differences of measurements equation for the carrier phase problem given the physical situation of m satellites with a known fixed receiver and a roving receiver whose position is to be estimated. The almost identical equation appears in van Graas and Lee [12, eq.(1)].

We can rewrite (2) as

$$y_k = \begin{bmatrix} e & E_k & I_m \end{bmatrix} \begin{bmatrix} \beta_k \\ x_k \\ N \end{bmatrix} + v_k, \tag{3}$$

and combine these for $k = 1, 2, \ldots$ to get

$$\begin{bmatrix} y_1 \\ \cdot \\ y_k \end{bmatrix} = \begin{bmatrix} e & E_1 & & & & I_m \\ & & \cdot & \cdot & & \cdot \\ & & & e & E_k & I_m \end{bmatrix} \begin{bmatrix} \beta_1 \\ x_1 \\ \cdot \\ \cdot \\ \beta_k \\ x_k \\ N \end{bmatrix} + \begin{bmatrix} v_1 \\ \cdot \\ v_k \end{bmatrix}, \tag{4}$$

where

$$\begin{bmatrix} v_1 \\ \cdot \\ v_k \end{bmatrix} \sim N(0, \sigma^2 I_{km}).$$

This is the mathematical model for which our positioning algorithm will be developed. Here for simplicity we have assumed that the number of visible satellites does

not change from one epoch to the next. Notice that in E_k, the unit vector e_k^i from the midpoint of the baseline x_k to satellite i depends on the baseline x_k. So we may write

$$E_k \equiv E(x_k).$$

This E_k is known once x_k is known. Given an approximation to x_k (our estimate of x_{k-1} say), we can compute our approximation to E_k. Then given the measurements y_k, we can estimate x_k, *etc.*.

3 A RECURSIVE LEAST SQUARES METHOD

In this section we use orthogonal transformation techniques to provide an efficient and numerically reliable method to recursively estimate the receiver positions and approximate the covariance matrices based on the model (4). We also discuss the requirement for the number of satellites in order to get meaningful position estimates.

3.1 Background for the LS solution

Suppose we have a linear model

$$y = Gz + v, \qquad v \sim N(0, \sigma^2 I), \tag{5}$$

where the matrix G is of full column rank. Then the best linear unbiased estimate (BLUE) of z is the solution of the LS problem

$$\min_z \|Gz - y\|^2.$$

A numerically stable approach to finding the LS solution is to compute the QR factorization of G

$$Q^T G \equiv \begin{bmatrix} U \\ W \end{bmatrix} G = \begin{bmatrix} R \\ 0 \end{bmatrix},$$

where $Q = [U^T, W^T]$ is orthogonal and usually is the product of Householder transformations or Givens rotation matrices, R is nonsingular upper triangular, and U has the same number of rows as R (see, for example [5, Ch.5]). Since the 2-norm is unaffected by orthogonal transformations, the LS solution $\hat{z}$ satisfies

$$\begin{aligned} \hat{z} &= \arg\min_z \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} z - \begin{bmatrix} Uy \\ Wy \end{bmatrix} \right\|^2 \\ &= \arg\min_z (\|Rz - Uy\|^2 + \|Wy\|^2), \end{aligned}$$

from which it follows

$$R\hat{z} = Uy.$$

This upper triangular system can be solved by back substitution. Thus

$$\hat{z} = G^\dagger y = R^{-1}Uy = R^{-1}U(Gz + v) = z + R^{-1}Uv,$$
$$\mathcal{E}\{\hat{z} - z\} = 0, \qquad \mathrm{cov}\{\hat{z} - z\} = \sigma^2 (R^T R)^{-1}. \tag{6}$$

We will also make use of the following:

$$\begin{aligned} &\mathcal{E}\{\|\hat{z} - z\|^2\} = \mathcal{E}\{\mathrm{trace}[(\hat{z} - z)(\hat{z} - z)^T]\} \\ &= \mathrm{trace}[\mathcal{E}\{(\hat{z} - z)(\hat{z} - z)^T\}] = \mathrm{trace}(\mathrm{cov}\{\hat{z} - z\}). \end{aligned} \tag{7}$$

3.2 The orthogonal transformation approach

For the time being we assume that the E_j ($j = 1, \ldots, k$) in (4) are known. In Remark 1 and Section 3.3 we will discuss how to compute them. The condition for (4) to have a unique LS solution is that the coefficient matrix has full column rank. Unfortunately it does not — the dependency comes from the first column and last m columns of the matrix in (3) being linearly dependent. One of the approaches to getting around the difficulty is to use the double differencing technique (see for example [8]), but this has some drawbacks. For example, it makes the measurements correlated. Here we use the most numerically reliable approach — orthogonal transformations of single differences.

Let P be an orthogonal transformation such that $P^T e = e_1 \sqrt{m}$. We could use some product of rotations to form P, but to be precise here we will use a Householder transformation (see for example [5, p.209]), which here has the form

$$P = I - \frac{2uu^T}{u^T u}, \qquad u = e_1 - e/\sqrt{m}.$$

Writing

$$[p_1, p_2, \ldots, p_m] \equiv [p_1, P_2] \equiv P, \tag{8}$$

and applying P^T to (2), we obtain the initial orthogonal transformation to (2)

$$\begin{bmatrix} p_1^T y_k \\ P_2^T y_k \end{bmatrix} = \begin{bmatrix} p_1^T E_k \\ P_2^T E_k \end{bmatrix} x_k + \begin{bmatrix} p_1^T N \\ P_2^T N \end{bmatrix} + e_1 \sqrt{m} \beta_k + \begin{bmatrix} p_1^T v_k \\ P_2^T v_k \end{bmatrix}. \tag{9}$$

Define

$$\begin{bmatrix} \delta^1 \\ \hline d \end{bmatrix} \equiv \begin{bmatrix} \delta^1 \\ \hline \delta^2 \\ \cdot \\ \delta^m \end{bmatrix} \equiv \begin{bmatrix} p_1^T N \\ \hline P_2^T N \end{bmatrix}, \qquad \gamma_k \equiv \delta^1 + \sqrt{m}\beta_k. \tag{10}$$

This eliminates one variable in (9) to give

$$\begin{bmatrix} p_1^T y_k \\ P_2^T y_k \end{bmatrix} = \begin{bmatrix} 1 & p_1^T E_k \\ & P_2^T E_k \quad I_{m-1} \end{bmatrix} \begin{bmatrix} \gamma_k \\ x_k \\ d \end{bmatrix} + \begin{bmatrix} p_1^T v_k \\ P_2^T v_k \end{bmatrix}, \quad (11)$$

$$\begin{bmatrix} p_1^T v_k \\ P_2^T v_k \end{bmatrix} \sim \mathcal{N}(0, \sigma^2 I_m).$$

Combining these for $k = 1, 2, \ldots$ and reordering gives

$$\begin{bmatrix} p_1^T y_1 \\ \cdot \\ p_1^T y_k \end{bmatrix} = \left[\begin{array}{ccc|ccc} 1 & & & p_1^T E_1 & & \\ & \cdot & & & \cdot & \\ & & 1 & & & p_1^T E_k \end{array}\right] \begin{bmatrix} \gamma_1 \\ \cdot \\ \gamma_k \\ x_1 \\ \cdot \\ x_k \end{bmatrix} + \begin{bmatrix} p_1^T v_1 \\ \cdot \\ p_1^T v_k \end{bmatrix}, \quad (12)$$

$$\begin{bmatrix} P_2^T y_1 \\ \cdot \\ P_2^T y_k \end{bmatrix} = \left[\begin{array}{ccc|c} P_2^T E_1 & & & I_{m-1} \\ & \cdot & & \cdot \\ & & P_2^T E_k & I_{m-1} \end{array}\right] \begin{bmatrix} x_1 \\ \cdot \\ x_k \\ d \end{bmatrix} + \begin{bmatrix} P_2^T v_1 \\ \cdot \\ P_2^T v_k \end{bmatrix}. \quad (13)$$

Notice that the transformed noise vectors in (12) and (13) follow the normal distributions $\mathcal{N}(0, \sigma^2 I_k)$ and $\mathcal{N}(0, \sigma^2 I_{k(m-1)})$, respectively. From the structures of the coefficient matrices in (12) and (13) we observe that in order to find the LS solution, the key is to find the LS solution of the reduced problem (13). Once the estimates of $x_1, \ldots, x_k$ are available from (13), we can easily get the estimates of $\gamma_1, \ldots, \gamma_k$, the combinations of the single differenced integer ambiguities and the single differenced receiver clock errors. But in practice, one may not be interested in these quantities.

Now look at the reduced problem (13) closely. Let B_k be the coefficient matrix in (13). Notice B_k is $k(m-1) \times (3k + m - 1)$. We obtain a unique LS solution if B_k has full column rank, which certainly requires $k(m-1) \geq 3k + m - 1$, i.e.,

$$m \geq 4 + \frac{3}{k-1}. \quad (14)$$

This relationship between the number of epochs and the required minimum number of satellites is displayed in the following table.

Table 1. Satellite requirements

# of epochs (k)	2	3	4	5	≥ 6
minimum # of satellites	7	6	5	5	5

We see that for position estimation, at least 5 satellites are required. Thus from now on we assume $m \geq 5$. It is easy to show when the column spaces of $P_2^T E_1, \ldots, P_2^T E_k$ do not intersect, then B_k has full column rank, see [2]. This is more likely as k increases. In the following we assume this is true as long as the number of epochs k and number of satellites m satisfy (14).

Let the QR factorization of $P_2^T E_j$ be

$$Q_j^T (P_2^T E_j) \equiv \begin{bmatrix} U_j \\ W_j \end{bmatrix} P_2^T E_j = \begin{bmatrix} R_j \\ 0 \end{bmatrix}, \quad (15)$$
$$3 \times (m-1) \;\; U_j, \quad (m-4) \times (m-1) \;\; W_j,$$

where $Q_j = [U_j^T, W_j^T]$ is an $(m-1) \times (m-1)$ orthogonal matrix, and R_j is a 3×3 nonsingular upper triangular matrix. Let

$$Q_j^T (P_2^T y_j) = \begin{bmatrix} U_j \\ W_j \end{bmatrix} P_2^T y_j \equiv \begin{bmatrix} u_j \\ w_j \end{bmatrix}, \quad (16)$$
$$Q_j^T (P_2^T v_j) = \begin{bmatrix} U_j \\ W_j \end{bmatrix} P_2^T v_j \equiv \begin{bmatrix} v_j^{(1)} \\ v_j^{(2)} \end{bmatrix}.$$

Transforming (13) by $\mathrm{diag}(Q_1^T, \ldots, Q_k^T)$ and reordering gives

$$\begin{bmatrix} u_1 \\ \cdot \\ u_k \end{bmatrix} = \left[\begin{array}{ccc|c} R_1 & & & U_1 \\ & \cdot & & \cdot \\ & & R_k & U_k \end{array}\right] \begin{bmatrix} x_1 \\ \cdot \\ x_k \\ d \end{bmatrix} + \begin{bmatrix} v_1^{(1)} \\ \cdot \\ v_k^{(1)} \end{bmatrix}, \quad (17)$$

$$\begin{bmatrix} w_1 \\ \cdot \\ w_k \end{bmatrix} = \begin{bmatrix} W_1 \\ \cdot \\ W_k \end{bmatrix} d + \begin{bmatrix} v_1^{(2)} \\ \cdot \\ v_k^{(2)} \end{bmatrix}. \quad (18)$$

Notice the second terms of the right hand sides of (17) and (18) follow the normal distributions $\mathcal{N}(0, I_{3k})$ and $\mathcal{N}(0, \sigma^2 I_{(m-4)k})$, respectively. We observe that d can be estimated by the LS solution of (18). When the LS estimate d_k for d is available, the $x_{1|k}, x_{2|k}, \ldots, x_{k|k}$, which are the estimates of $x_1, x_2, \ldots, x_k$ at epoch k, can be computed from (17) by solving the upper triangular systems

$$R_j x_{j|k} = u_j - U_j d_k, \qquad j = 1, \ldots, k. \quad (19)$$

Notice that the $x_{1|k}, x_{2|k}, \ldots, x_{k|k}$ can be computed in any order once d_k is available. So if d_k is updated, we could for example update $x_{k|k}$ without updating any of the earlier position estimates. Since the coefficient matrix in (18) is $k(m-4) \times (m-1)$, in order to get a unique LS solution for d we certainly require

$$k(m-4) \geq m - 1,$$

which is equivalent to (14).

Now our problem is to obtain the estimate of d from (18). We use a recursive approach. Suppose at epoch $k-1$, the corresponding coefficient matrix in (18) has full column rank (in the initial stage the coefficient matrix does not have full column rank and we will discuss

this case later in Section 3.3). We also assume we have obtained the following orthogonal transformations:

$$T_{k-1}^T \begin{bmatrix} W_1 \\ \cdot \\ W_{k-1} \end{bmatrix} = \begin{bmatrix} S_{k-1} \\ \hline 0 \end{bmatrix}, \quad T_{k-1}^T \begin{bmatrix} w_1 \\ \cdot \\ w_{k-1} \end{bmatrix} = \begin{bmatrix} \hat{w}_{k-1} \\ \hline \bar{w}_{k-1} \end{bmatrix},$$

where T_{k-1} is orthogonal, S_{k-1} is nonsingular upper triangular and has the same number of rows as $\hat{w}_{k-1}$. Then at epoch k after obtaining E_k and the QR factorization of $P_2^T E_k$, we perform the following orthogonal transformations:

$$\tilde{T}_k^T \begin{bmatrix} S_{k-1} \\ W_k \end{bmatrix} = \begin{bmatrix} S_k \\ \hline 0 \end{bmatrix}, \tag{20}$$

$$\tilde{T}_k^T \begin{bmatrix} \hat{w}_{k-1} \\ w_k \end{bmatrix} = \begin{bmatrix} \hat{w}_k \\ \hline \bar{\bar{w}}_k \end{bmatrix}, \quad \bar{w}_k \equiv \begin{bmatrix} \bar{w}_{k-1} \\ \bar{\bar{w}}_k \end{bmatrix}, \tag{21}$$

where $\tilde{T}_k$ is orthogonal, S_k is nonsingular upper triangular, S_k and $\hat{w}_k$ have the same number of rows. The orthogonal transformations can be implemented by a sequence of Householder transformations, which make use of the upper triangular structure of S_{k-1}. But the matrices $\tilde{T}_k$ and T_{k-1} are neither formed nor stored. Therefore we get the transformed form of the LS problem (18):

$$\begin{bmatrix} S_k \\ 0 \end{bmatrix} d \approx \begin{bmatrix} \hat{w}_k \\ \bar{w}_k \end{bmatrix}, \tag{22}$$

where for simplicity we have omitted the transformed noise vector, which still follows the normal distribution $\mathcal{N}(0, \sigma^2 I_{(m-1)k})$ and is not used in our computation. Thus by solving the upper triangular system

$$S_k d = \hat{w}_k, \tag{23}$$

we obtain d_k, the estimate of d at epoch k. After this, we can solve (19) to obtain the $x_{j|k}$, the estimate of x_j at epoch k.

Remark 1. Since $E_k = E(x_k)$ $(k = 1, 2, \ldots,)$, our problem of estimating the positions is actually nonlinear. We have to use approximations to E_k during the processing. Suppose we have obtained an estimate $x_{k-1|k-1}$ of x_{k-1}. Then we use $E(x_{k-1|k-1})$ as an approximation of E_k. This approximation is usually acceptable, since usually x_{k-1} and x_k are not far away from each other, and E_{k-1} and E_k are very close. Also if necessary, after obtaining the estimate $x_{k|k}$ of x_k, we can use $E(x_{k|k})$ to approximate E_k, and we could even do some further iterations to get an improved estimate of x_k. But our preliminary tests suggest the iterations do not bring any significant improvement.

Remark 2. From (22) we observe $\|\bar{w}_k\|$ is the least squares residual of (2). This information is useful for fault detection, see [11, Chap. 7].

3.3 Computing the initial points

This section deals with the initial stage, which is from epoch $k = 1$ until some epoch such that at the end of this stage the coefficient matrix in (18) reaches full column rank. The estimate from the least squares model is not unique until this point, and in ordinary linear least squares problems is usually not needed or computed. However in this slightly nonlinear case E_k depends on x_k, and so we need to provide estimates of these initial x_k.

At the beginning ($k = 1$), we do not know x_1 (since this is what we want). But in many GPS applications we may know an approximate location of the roving receiver (in fact we can often use code measurements to estimate the initial position). Then we can use this to construct an approximation to E_1. If we do not have any information about the position of the roving receiver and do not bother to use code measurements, we may take each e_1^i in E_1 to be the direction cosine from the *stationary* receiver to satellite i, for $i = 1, \ldots, m$, see (1), in other words, we may take $x_{1|1} = 0$.

Suppose we have already obtained $x_{j|j}$, the estimate of x_j at epoch j. We would like to get an estimate of x_{j+1} when the measurements at epoch $j + 1$ are available. We use the idea of [12]. From (13) we obtain

$$P_2^T E_{j+1} x_{j+1} = P_2^T E_j x_j + P_2^T (y_{j+1} - y_j) - P_2^T (v_{j+1} - v_j).$$

Notice $P_2^T (v_{j+1} - v_j) \sim N(0, 2\sigma^2 I)$. Solving the LS problem

$$P_2^T E_{j+1} x_{j+1|j+1} \approx P_2^T E_j x_{j|j} + P_2^T (y_{j+1} - y_j)$$

for $x_{j+1|j+1}$, our estimate for x_{j+1}, gives

$$x_{j+1|j+1} = (P_2^T E_{j+1})^\dagger [P_2^T E_j x_{j|j} + P_2^T (y_{j+1} - y_j)]. \tag{24}$$

Notice E_{j+1} is actually unknown, but we can take $E_{j+1} = E_j$, and with the QR factorization of $P_2^T E_j$ in (15), we get the preliminary estimate

$$x_{j+1|j+1} = x_{j|j} + R_j^{-1} U_j P_2^T (y_{j+1} - y_j).$$

Then we can use this $x_{j+1|j+1}$ to construct E_{j+1} and compute its QR factorization. From (24), we have the new estimate

$$x_{j+1|j+1} = R_{j+1}^{-1} U_{j+1} [P_2^T E_j x_{j|j} + P_2^T (y_{j+1} - y_j)].$$

Of course we do not compute R_j^{-1} and R_{j+1}^{-1} in the above computations. Instead we solve upper triangular systems.

Now we could use this new estimate to update E_{j+1} again and compute its QR factorization, which would

be used in the next epoch. But our preliminary tests show that there is no significant difference if we do not update E_{j+1} and re-compute the QR factorization.

We continue the above process until the coefficient matrix in (18) is of full column rank, i.e., (18) has a unique LS estimate for d. When epoch k and the number of satellites m satisfy the inequality (14), the coefficient matrix usually has full column rank. For example, if there are 6 satellites available, then after 3 epochs the coefficient matrix usually becomes full column rank.

Suppose after k epochs, the coefficient matrix is of full column rank. Then do the following orthogonal transformations (QR factorization)

$$T_k^T \begin{bmatrix} W_1 \\ \cdot \\ W_k \end{bmatrix} = \begin{bmatrix} S_k \\ \hline 0 \end{bmatrix}, \quad T_k^T \begin{bmatrix} w_1 \\ \cdot \\ w_k \end{bmatrix} = \begin{bmatrix} \hat{w}_k \\ \hline \bar{w}_k \end{bmatrix},$$

where W_j and w_j $(1 \le j \le k)$ were obtained in the j-th step (see (15) and (16)), T_k is orthogonal, S_k is nonsingular upper triangular and has the same number of rows as $\hat{w}_j$. The orthogonal transformations can be implemented by a sequence of Householder transformations. But T_k is neither formed nor stored. With S_k and $\hat{w}_k$, we go to the regular stage, see Section 3.2.

3.4 Approximating covariance matrices

In order to have some idea of the errors in the estimates of baselines and the estimates of transformed integer ambiguities ($d = P_2^T N$, see (10)), we would like to know the corresponding covariance matrices. The orthogonal transformations (including permutations corresponding to reordering) transformed the coefficient matrix in (12) and (13) to the following upper triangular matrix (see (17), (18), and (22)),

$$R \equiv \left[\begin{array}{ccc|c} 1 & & & p_1^T E_1 & & & \\ & 1 & & & p_1^T E_2 & & \\ & & \cdot & & & \cdot & \\ & & 1 & & & p_1^T E_k & \\ \hline & & & R_1 & & & U_1 \\ & & & & R_2 & & U_2 \\ & & & & & \cdot & \cdot \\ & & & & & R_k & U_k \\ \hline & & & & & & S_k \end{array}\right].$$

Its $2k+1$ block columns correspond to the unknowns $\gamma_1, \gamma_2, \ldots, \gamma_k, x_1, x_2, \ldots, x_k, d$, respectively (each block column corresponding to a γ_j actually has only one column, each block column corresponding to an x_j has 3 columns, and the last block column corresponding to d has $m-1$ columns).

If (4) was an exact *linear* model, then we could obtain the exact covariance matrices, see Section 3.1. But since E_k depends on x_k, the above matrix R depends on the unknowns, so here we will only approximate the true covariance matrices. To do so we assume R does not depend on the unknowns, since this appears to give acceptable results, see Section 4. Then according to (6), $\mathrm{cov}\{d_k - d\}$ is σ^2 times the bottom right hand corner block of $(R^T R)^{-1}$, which is just $S_k^{-1} S_k^{-T}$, i.e.,

$$\mathrm{cov}\{d_k - d\} = \sigma^2 S_k^{-1} S_k^{-T} = \sigma^2 (S_k^T S_k)^{-1}. \qquad (25)$$

Thus S_k/σ is the Cholesky factor of $[\mathrm{cov}\{d_k - d\}]^{-1}$. This is very nice, since the inverse of a covariance matrix is more useful than the covariance matrix itself in many cases, and for numerical reliability it is better to work with the Cholesky factor of a positive definite matrix rather than the matrix itself.

Now for $j = 1, \ldots, k$ we would like to get the covariance matrix $\mathrm{cov}\{x_{j|k} - x_j\}$, which, according to (6), is the $(k+j)$-th 3×3 block on the diagonal of $\sigma^2 (R^T R)^{-1}$. In order to obtain that, we apply an orthogonal matrix Z_j to R from the left to zero the U_j in R by using only the $(k+j)$-th block row in which R_j and U_j lie and the last block block row in which S_k lies. Essentially we compute an orthogonal matrix $\bar{Z}_j$ such that

$$\bar{Z}_j^T \begin{bmatrix} R_j & U_j \\ 0 & S_k \end{bmatrix} = \begin{bmatrix} \bar{R}_j & 0 \\ \tilde{R}_j & \tilde{S}_k \end{bmatrix},$$

where $\bar{R}_j$ is upper triangular. The computation can be done by Givens rotations to make full use of the upper triangular structure of R_j and S_k, for more details, see [9, 2]. Since the $(k+j)$-th block row in $Z_j^T R$ now has the form

$$[0, \cdots, 0, \bar{R}_j, 0, \cdots, 0], \qquad (26)$$

it is easy to verify the $(k+j)$-th block row in $(Z_j R)^{-1}$ has the form

$$[0, \cdots, 0, \bar{R}_j^{-1}, 0, \cdots, 0].$$

Therefore the $(k+j)$-th 3×3 block on the diagonal of $(Z_j^T R)^{-1}(Z_j^T R)^{-T}$ is just $\bar{R}_j^{-1}\bar{R}_j^{-T}$. But since

$$(R^T R)^{-1} = [(Z_j R)^T Z_j^T R]^{-1} = (Z_j^T R)^{-1}(Z_j^T R)^{-T},$$

it follows that

$$\mathrm{cov}\{x_{j|k} - x_j\} = \sigma^2 \bar{R}_j^{-1}\bar{R}_j^{-T} = \sigma^2 (\bar{R}_j^T \bar{R}_j)^{-1}.$$

Thus $\bar{R}_j/\sigma$ is the Cholesky factor of $[\mathrm{cov}\{x_{j|k} - x_j\}]^{-1}$.

Similarly, but more simply, we can compute $\mathrm{var}\{\gamma_{j|k} - \gamma_j\}$ for $j = 1, \ldots, k$. But one may not be interested in these.

4 EXPERIMENTAL RESULTS

In order to demonstrate the performance of our algorithm, we first give some computer simulation results. All our computations were performed in MATLAB 5.2 on a Pentium III running Windows 2000. A 24 GPS satellite constellation was used in the simulations. The roving receiver is assumed to be on board an aircraft circling above a reference station at the constant speed of $100\,m/s$. The baseline is about 1 km. Each single differenced carrier phase measurement is corrupted by a random normally distributed noise with zero mean and standard deviation $\sigma = 0.002\,m$. The receiver clock offset relative to GPS time is modeled by white noise input to a second order Markov process based on [10, p.417]. The time interval between two consecutive epochs is 1 second.

The typical results for 7 satellites with initial errors of 5 meters, 100 meters, 1 kilometer and 10 kilometers are shown in Figures 1–4, respectively. The typical results for 6 and 5 satellites with an initial error of 100 meters are shown in Figures 5 and 6. From these figures we see the new algorithm performs better than the van Graas and Lee algorithm in terms of position accuracy at the beginning. But later the two algorithms may not have any significant differences, particularly when 7 satellites are used. However the new algorithm is more efficient than the van Graas and Lee algorithm. Our MATLAB simulations show for each epoch, the number of floating point operations of the former is about half that of the latter. Figure 5 shows that using 6 satellites takes more time to get submeter accuracy than using 7 satellites. Figure 6 shows that with 5 satellites it takes much longer still to get submeter accuracy.

In order to test our covariance approximations from Section 3.4, we use the fact that for a linear system $\mathcal{E}\{\|\hat{z}-z\|^2\} = \text{trace}(\text{cov}\{\hat{z}-z\})$ (see (7)). Thus we can test the trace of our approximate covariance matrix by plotting the position (or baseline) errors $\|x_{k|k} - x_k\|$ and $(\text{trace}(\text{cov}\{x_{k|k} - x_k\})^{1/2}$ for our approximation. This is done in Figure 7, where 7 satellites are used for positioning with 1 km initial error. From this figure, we see our approximate variances (we are only considering the diagonal of the covariance matrix) appear to give reasonable indicators of the position errors. Note this is a different run with different noise to that in Figure 3.

We also did some preliminary real data tests. The data set was provided by VIASAT Geo-Technologies Inc. in Montreal, and was recorded on April 1, 2000. The user was walking in an open sky environment with 7 satellites visible. The initial baseline was about 130 m. Because we did not know the true positions, we followed standard practice and regarded the true positions to be the the position estimates obtained by a software package of VIASAT Technologies Inc. which used all available information from their single frequency receiver to correctly fix the ambiguities. Our initial computations again suggest that our algorithm performs better than the van Grass and Lee algorithm. However we have not had time to validate these computations, and hope to publish the fully verified results later, see [2].

Position errors for 7 satellites with different initial errors (Figure 1–Figure 4)

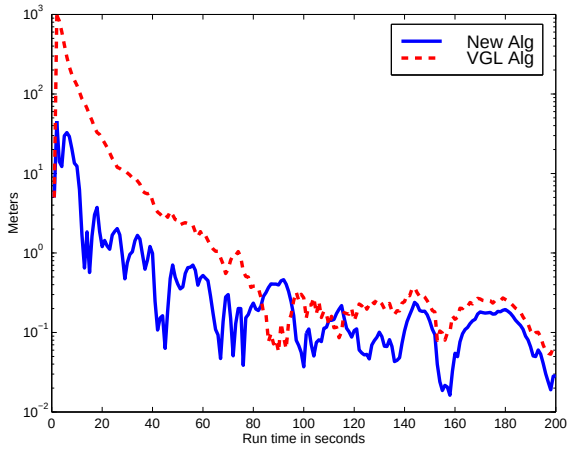

Figure 1: 5 m initial error

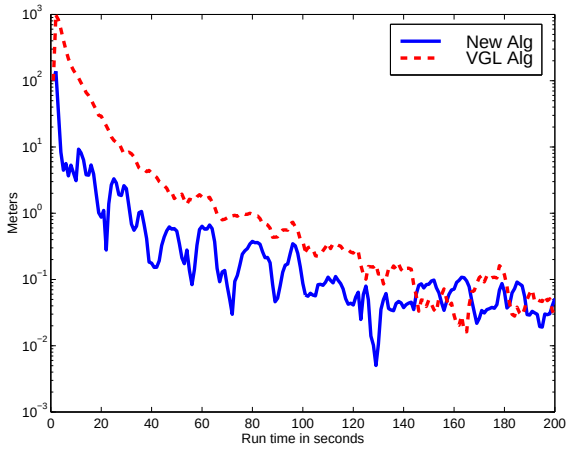

Figure 2: 100 m initial error

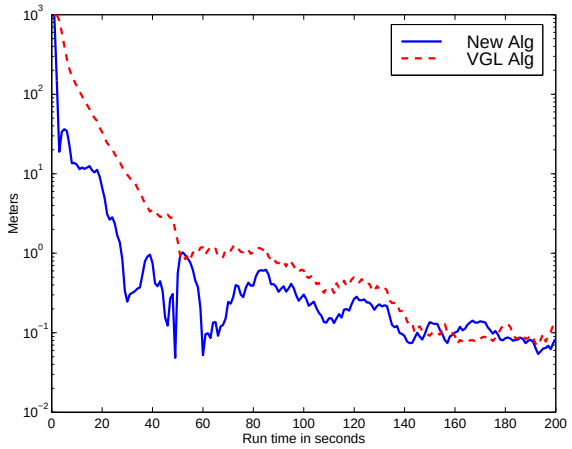

Figure 3: 1 km initial error

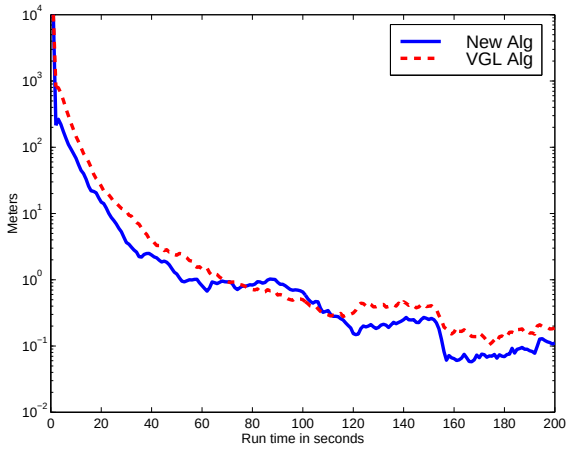

Figure 4: 10 km initial error

Position errors for 6 and 5 satellites with 100 m initial error (Figure 5–Figure 6)

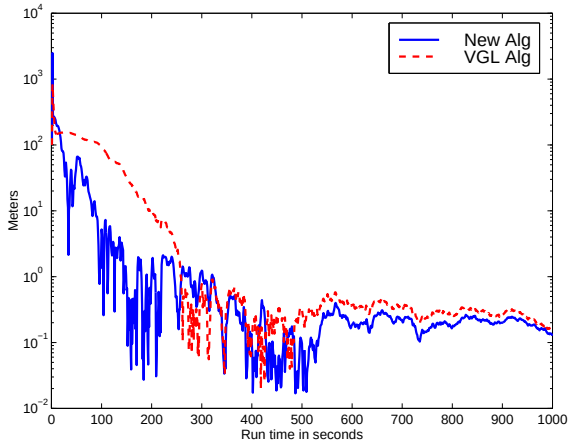

Figure 5: 6 satellites

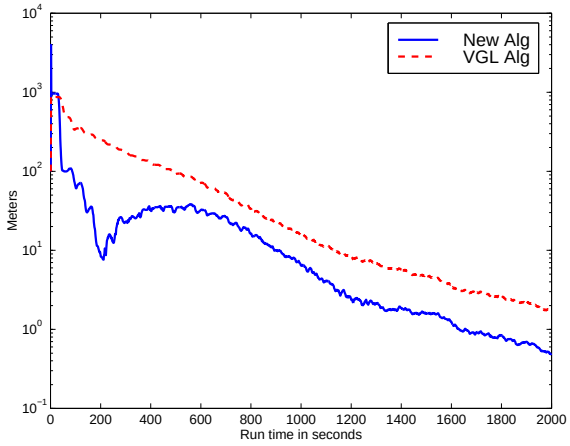

Figure 6: 5 satellites

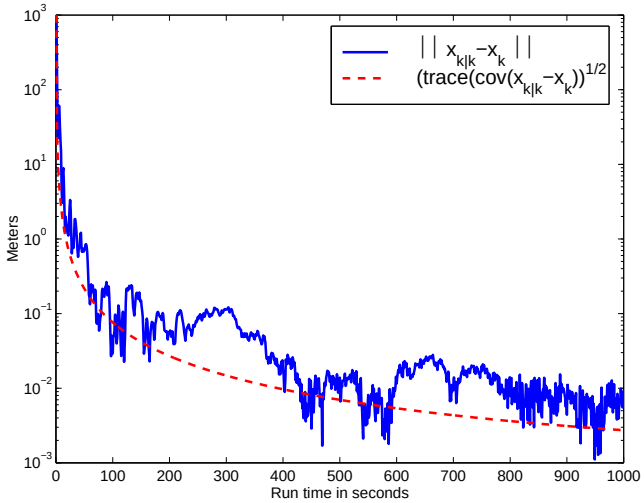

Figure 7: Position errors and approximate variances with 7 satellites and 1 km initial error

5 Remarks and Conclusions

Notice that in our algorithms we did not fix the integer ambiguities. There are two reasons. One is that for some applications, the accuracy of position estimates obtained with non-integer ambiguities may be enough. The other is that if one wants, ambiguities can be fixed as integers by using the covariance matrix of d_k in (25).

In our algorithm, we assume we always have the same visible satellites. But this may not be true in practice when the time span for navigation is long. It is not difficult to modify our algorithm to handle this situation. We hope to deal with the above two topics in [2].

When we apply our algorithm to real data, we need to incorporate an algorithm for cycle slip detection and correction. This can be done without much difficulty.

Our algorithm works in theory when only 5 satellites are visible. But like the van Graas and Lee algorithm, the initial time to get submeter accuracy is long.

From the preliminary experimental results we see that the recursive LS approach is effective, and that it is not necessary to use the Kalman filter for position estimation, unless perhaps we have information about the velocity of the receiver. Since we always use numerically stable orthogonal transformations, from a numerical point of view our implementation is numerically reliable. Our algorithm is also efficient, since it takes full advantage of the structure of the problem.

ACKNOWLEDGMENTS

This research was supported by NSERC of Canada Grant RGPIN217191-99 and FCAR of Quebec, Canada Grant 2001-NC-66487 for X.-W. Chang, and by NSERC of Canada Grant OGP009236 for C. C. Paige. The authors would like to thank Dr. Jianjun Zhu of Laval University and Mr. Claude St-Pierre of VIASAT Technologies Inc for their help in the real data test.

References

[1] J. Ashjaee, GPS Doppler Processing for Precise Positioning in Dynamic Applications, *Navigation, Journal of The Institute of Navigation*, Vol. 32, No. 4, Winter 1985-86.

[2] X.-W. Chang and C. C. Paige, An efficient and numerically reliable algorithm for GPS positioning. In preparation.

[3] C. C. Counselman, III and S. A. Courevitch, Miniature Interferometer Terminals for Earth Surveying: Ambiguity and Multipath with Global Positioning System, *IEEE Transaction on Geoscience and Remote Sensing*, Vol. GE-19, No. 4, 1981.

[4] A. G. Evans, B. R. Hermann and P. J. Fell, Global Positioning System Sensitivity Experiment, *Navigation, Journal of The Institute of Navigation*, Vol. 28, No. 2, Summer 1981.

[5] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University press, 3nd ed., Baltimore, MD, 1996.

[6] P. Y. C. Hwang, Kinematic GPS for Differencing Positioning: Resolving Integer Ambiguities on the Fly, *Navigation, Journal of The Institute of Navigation*, Vol. 38, No. 1, Spring 1991.

[7] P. V. W. Loomis, A kinematic GPS Double Differencing Algorithm, *Proceeding of the Fifth International Geodetic Symposium on Satellite Positioning*, New Mexico State University, Las Cruces, NM, March 13-17, 1989.

[8] W. Michalson, H. Hua, and D. Cox, GPS Carrier-Phase RAIM, *Proceedings of ION GPS-98*, Nashville, Tennessee, 1998, pp. 1975–1984.

[9] C. C. Paige and M. A. Saunders, Least Squares Estimation of Discrete Linear Dynamic System Using Orthogonal Transformations, *SIAM J. Numer. Anal.*, Vol. 14, 1977, pp. 180–193.

[10] B.W. Parkinson, J.J. Spilker, P. Axelrad, P. Enge (ed), Global Positioning System: Theory and Applications, Vol 1 and 2, American Institute of Aeronautics and Astronautics, Inc, 1996.

[11] P. J. Teunissen and A. Kleusberg (Eds). *GPS for Geodesy*, 2nd ed. Springer, Berlin, 1998.

[12] F. van Graas and S.-W. Lee, High-Accuracy Differential Positioning for Satellite-Based Systems without Using Code-Phase Measurements, *Navigation, Journal of The Institute of Navigation*, Vol 42, Winter 1995, pp. 605–618.