Design and Analysis of a New GPS Algorithm

Wei Li[1#] Deng Li[2] Shuhui Yang[3] Zhiwei Xu[4] Wei Zhao[1*]

[1] *University of Macau*
[2] *Rensselaer Polytechnic Institute*
[3] *Purdue University Calumet*
[4] *Institute of Computing Technology, CAS*
liwei@ict.ac.cn, dengl4@rpi.edu, yang246@purdue.edu, zxu@ict.ac.cn, weizhao@umac.mo

Abstract

In this paper, we propose and analyze a new GPS positioning algorithm. Our algorithm uses the direct linearization technique to reduce the computation time overhead. We invoke the general least squares method in order to achieve optimality in the situation when the trilateration system of equations becomes over-determined. We systematically evaluate our new algorithms and show that they indeed take much less computation time than the traditional GPS method while maintaining reasonable accuracy.

1. Introduction

Global Positioning System (GPS) is a space-based system, providing worldwide location and navigation services [14] [15]. Though GPS is a huge and sophisticated system, its basic idea comes from a fundamental mathematical theorem that in a three-dimensional space we can locate the position of an object if we know the coordinates of three other points and their distances from the object. Such a method is commonly referred to as *Trilateration* [20]. The nature of trilateration is to solve a multivariate quadratic equation system. Different methods [2] [11] [22] [23] [26] have been proposed to find a solution.

Generally, solutions for the trilateration problem need to address two issues: the first is to construct the pseudo-range equation model. Based on different assumptions or conditions, various models can be established. For example, in *P4P* (*Pseudo-ranging 4-point Problem*) model [13], clock bias is treated as an unknown parameter. Then, using such a model to solve the trilateration problem requires at least four satellites. The second issue is how to solve the equation systems built by pseudo-range equation models. A typical approach is to use the *iterative method*. Because pseudo-range equation models are non-linear, iterative methods are natural choices in GPS positioning. However, iterative methods have shortages of expensive computational overhead and risk of non-convergence.

However, in many application systems, the object to be positioned may move at a high speed. It is then necessary to reduce the computation time overhead in order to provide real-time response for positioning requests.

In this paper, our goal is to develop new GPS positioning algorithms that will make a significant reduction on computation overhead while maintaining comparable positioning accuracy with traditional GPS algorithms. In particular, we take a *direct linearization* approach that allows us to remove quadratic items, hence forming a linear system. Therefore, our method can deliver a closed-form solution. Such a solution obviously uses less computation time than traditional iterative methods.

We systematically analyze and compare the performance of the proposed approach with traditional iterative methods with real data sets from several GPS stations which are currently in operation. We find that our new methods indeed reduce the computation time overhead significantly when compared with the traditional iterative method. Meanwhile, our methods are able to maintain an accuracy level which is sufficiently reasonable for practical usage.

The paper is organized as follows: Section 2 discusses the previous work; Section 3 models the trilateration problem by presenting and analyzing a typical iterative method. Section 4 introduces our method in detail.

* Corresponding author.
# On leave from the *Institute of Computing Technology, CAS.*

1063-6927/10 $26.00 © 2010 IEEE
DOI 10.1109/ICDCS.2010.53

Section 5 shows the experimental results, and Section 6 provides the final remarks.

2. Related Work

Fundamentally, solving the GPS positioning problem involves two issues: first, to establish a proper pseudo-range model; second, to choose an optimal or near-optimal method to solve the model which is typically non-linear.

For the first issue, various pseudo-range models have been developed, based on different assumptions. A commonly used model is called *P4P* (*Pseudo-ranging 4-point Problem*) [13]. This model treats the receiver's coordinates and clock bias as four unknown parameters. A basic assumption of this model is that GPS receivers use a low-cost clock which can not be fully synchronized with the standard time. Previous studies reported in [1], [2], [4], [11], [22], [23] and [26] adopted this kind of model.

When precise clock time can be acquired, only three satellites are needed to calculate a position. In [30] an investigation is made on the feasibility of using this kind of idea to solve the problem. [27] further pointed out that precise clock time could bring additional benefits on vertical position accuracy. As such, many previous studies have considered using the c*lock bias prediction* to provide a good estimation on true time [3] [17] [27] [34] [35] [36].

Two different methods have been taken to address the second issue: *direct methods* and *iterative methods*. Previous studies on direct solutions included those reported in [1] [2] [6] [18] [19] [22] [23]. The iterative solutions are covered in [4] [11] [25] [26] [28]. Generally, direct methods can obtain closed-form solutions and need less execution time. Direct methods can also avoid the problems of bad initial guess and non-convergence which exist in iterative methods. However, most of the direct methods proposed in previous studies have assumed that the pseudo-range equation system is deterministic [28]. This unrealistic assumption makes most of direct methods not practical in real applications [26]. Furthermore, some direct methods such as [1] [18] can only work with four satellites.

On the contrary, the traditional iterative solution has been widely used in current GPS devices. *Newton-Raphson Method* (NR for abbreviation) is mostly used in practice [11]. Though an iterative method typically requires more execution time, it can tolerate measurement errors and support the case when there are more than four satellites. Bad initial guess and failure to converge may pose potential problems for iterative methods.

The method we proposed in this paper is very different from aforementioned methods. First, our method treats unknown clock bias as a correctable error contained in pseudo-range. Second, we provide a direct linearization method to convert the pseudo-range equation model to a linear one. Then, a closed-form solution can be derived[1]. In the following sections, we will introduce our method in detail and analyze and compare it with methods proposed in the previous studies.

3. Classic GPS Method

3.1. A Theoretical Model of Trilateration

The Global Positioning System (GPS) is a global navigation satellite system consisting of the following segments [24]:

1) **The space segment** of at least 24[2] space vehicles (i.e., satellites) orbiting in 6 circular orbital planes around the earth;
2) **The user segment** of unlimited GPS ground receiver; and
3) **The control segment** of several base stations on the earth that are monitoring and maintaining the satellites.

The goal of the system is to estimate the locations of GPS ground receivers. Without any consideration of obstacles, a GPS ground receiver at any point on the earth can receive signals from 6 to 10 (or more) different GPS satellites based on the RF line-of-sight propagation. The positioning process is performed according to the basic trilateration method:

- A receiver obtains a signal from satellite S_i, and uses the information from the signal to compute the distance between itself and S_i . Let this distance be ρ_i . Let S_i's coordinates be (x_i, y_i, z_i), and the receiver's location (to be estimated) be (x^e, y^e, z^e), The center of the earth is the origin $(0,0,0)$. Then, the following distance equation can be established:

[1] Because GPS positioning depends on measuring travel time of signals, it can be seen as a kind of TOA-based localization [31] [9] [7] [5]. A similar direct linearization method was proposed in [A]. The differences between our solution and the method in [A] are clock bias prediction model and an optimal algorithm, discussed in Section 4.

[2] In March 2008, there were 31 active satellites.

$$\sqrt{(x_i - x^e)^2 + (y_i - y^e)^2 + (z_i - z^e)^2} = \rho_i \quad (3\text{-}1)$$

- By using signals from three different satellites (say, S_1, S_2, and S_3), we will have a system of equations as follows:

$$\sqrt{(x_1 - x^e)^2 + (y_1 - y^e)^2 + (z_1 - z^e)^2} = \rho_1 \quad (3\text{-}2)$$

$$\sqrt{(x_2 - x^e)^2 + (y_2 - y^e)^2 + (z_2 - z^e)^2} = \rho_2 \quad (3\text{-}3)$$

$$\sqrt{(x_3 - x^e)^2 + (y_3 - y^e)^2 + (z_3 - z^e)^2} = \rho_3 \quad (3\text{-}4)$$

Solving this system of equations, we can determine $\left(x^e, y^e, z^e\right)$, an estimation of the receiver's location. Note that theoretically, solutions for these kinds of equations may not be unique, but the physical meaning of the equations usually results in only one solution. The geometric meaning of the method can be found in [20].

3.2. An Error Model of Distance Estimation

In the above theoretical model, an implicit assumption is made. That is, there is no error in estimating the distance between the receiver and a satellite. This assumption may not be realistic in the real world. Indeed, in reality, the distance estimated can be written in the form as follows:

$$\rho_i^e = \rho_i + \varepsilon_i^S + \varepsilon^R \quad \text{or} \quad \rho_i = \rho_i^e - (\varepsilon_i^S + \varepsilon^R) \quad (3\text{-}5)$$

where ρ_i is the exact distance between the receiver and S_i, ε^R is the receiver dependent error, and ε_i^S is satellite dependent error.

Let us use an example to illustrate factors in a real system that may contribute to ε_i^S and ε^R. Let s be the time when the satellite sends a signal and t be the time when the receiver receives the signal. The exact distance between satellite i and the receiver is then given by

$$\rho_i = (s - t)c_i^a \quad (3\text{-}6)$$

where c_i^a is the average speed at which the signal travels from S_i to the receiver. If the clocks of the satellite and the receiver are perfectly synchronized and c_i^a is known, the above computation would be accurate.

However, in reality, it is not the case. First, it is unrealistic to assume that the clock at the receiver can precisely synchronize with those of the satellites as it would be too costly to realize synchronization at a reasonable level of accuracy[3]. As such, we may have to use t^e, the reading of the receiver's clock at time t, as an estimate of t. Let the error of the clock be Δt. Then, we have

$$t^e = t + \Delta t \quad (3\text{-}7)$$

Second, when the radio signal travels from a satellite to the receiver, it passes through different media and hence the speed may vary. We would not be able to know the accurate average speed at which the signal travels. We may estimate c_i^a by using c, the speed when light travels in a vacant space. This estimate, of course, will result in an error:

$$\Delta c_i = c_i^a - c \quad (3\text{-}8)$$

Now, an estimate of the distance between the satellite and the receiver is given by:

$$\rho_i^e = (s - t^e)c \quad (3\text{-}9)$$

$$= (s - t - \Delta t)(c_i^a - \Delta c_i) \quad (3\text{-}10)$$

$$= (s - t)c_i^a - c\Delta t - (s - t)\Delta c_i \quad (3\text{-}11)$$

Comparing (3-11) with (3-6), (3-7), and (3-8), the first term in (3-11) is ρ_i, the accurate distance. The second term is, ε^R, the receiver dependent error, and the third term in (3-11) is ε_i^S, the satellite dependent error. While there are other factors that may contribute to the errors, this example confirms that (3-6) is a proper model for distance estimation and relevant errors.

3.3. Dealing with Clock Dependent Error

If we substitute (3-5) into (3-2), (3-3), and (3-4), then the system of three equations has 7 unknown variables, namely $\left(x^e, y^e, z^e, \varepsilon_1^S, \varepsilon_2^S, \varepsilon_3^S, \varepsilon^R\right)$. Clearly, three equations are insufficient to solve the system of equations in order to determine 7 unknowns. An approach is to use more satellites and hence set up more equations.

In the case where there are only clock dependent errors, or where satellite dependent errors can be compensated, 4 satellites are sufficient. For example, *Differential GPS (DGPS)* technology as described in [24] [29] can be used. In this case, we have the following system of equations:

$$\sqrt{(x_1 - x^e)^2 + (y_1 - y^e)^2 + (z_1 - z^e)^2} = \rho_1^e - \varepsilon^R \quad (3\text{-}12)$$

$$\sqrt{(x_2 - x^e)^2 + (y_2 - y^e)^2 + (z_2 - z^e)^2} = \rho_2^e - \varepsilon^R \quad (3\text{-}13)$$

[3] The satellites usually have an atomic clock that can be synchronized up to an accuracy of 10^{-15} second, sufficient for the location calculation. Unfortunately, a civil mobile device can not afford such an expensive atomic clock.

$$\sqrt{(x_3-x^e)^2+(y_3-y^e)^2+(z_3-z^e)^2}=\rho_3^e-\varepsilon^R \quad (3\text{-}14)$$

$$\sqrt{(x_4-x^e)^2+(y_4-y^e)^2+(z_4-z^e)^2}=\rho_4^e-\varepsilon^R \quad (3\text{-}15)$$

Solving the above system of equations, one may determine (x^e, y^e, z^e), an estimate of receiver's location, and ε^R, the receiver dependent error.

Unfortunately, this approach of using more satellites can not be directly generalized into the case when both types of errors exist. Consider the following argument: **s**ubstituting (3-5) into (3-1), we have

$$\sqrt{(x_i-x^e)^2+(y_i-y^e)^2+(z_i-z^e)^2}=(\rho_i^e-\varepsilon_i^S)-\varepsilon^R \quad (3\text{-}16)$$

where $i=1,\ 2,\ \ldots,\ m$ and $m\geq 4$. For this system of m equations, there will be $3+m+1$ unknown variables and hence the system of equations cannot be directly solved. A much more sophisticated approach must be taken, as we will discuss in the next subsection.

3.4. The Newtown-Raphson Method

In this sub-section, we describe that the Newtown-Raphson (NR) method can deal with the receiver dependent error and has been utilized in many GPS devices. This method forms a foundation for the algorithms we will propose in Section 4.

3.4.1. The Algorithm

As the system of equations in (3-16) is not directly solvable, the NR algorithm takes an approximation approach by treating ε_i^S as an error in computation of ρ_i^e and hence assumes that it can be ignored in the solution process. Thus, the NR method really solves the following system of equations

$$\Re_i \approx \rho_i^e-\varepsilon^R \quad (3\text{-}17)$$

where

$$\Re_i=\sqrt{\left((x_i-x^e)^2+(y_i-y^e)^2+(z_i-z^e)^2\right)} \quad (3\text{-}18)$$

where $i=1,\ 2,\ \ldots,\ m$ and $m\geq 4$.

We need to introduce some notations first. At the k-th iteration, let $\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)$ be the values of $\left(x^e,y^e,z^e,\varepsilon^R\right)$ and $\Re_i^{[k]}$ be the value of $\Re_i$. Define the residual function $P_i\left(x^e,y^e,z^e,\varepsilon^R\right)$ as follows:

$$P_i=\Re_i-\rho_i^e+\varepsilon^R \quad (3\text{-}19)$$

Define partial derivatives of P_i as follows:

$$X'_i=\partial P_i/\partial x^e \quad (3\text{-}20)$$

$$Y'_i=\partial P_i/\partial y^e \quad (3\text{-}21)$$

$$Z'_i=\partial P_i/\partial z^e \quad (3\text{-}22)$$

$$\mathrm{E}'_i=\partial P_i/\partial \varepsilon^R \quad (3\text{-}23)$$

The algorithm is an iterative one. We define residual function at the k-th iteration as follows:

$$P_i^{[k]}=\Re_i^{[k]}-\rho_i^e+\varepsilon^{R[k]} \quad (3\text{-}24)$$

In order to find a solution close enough to the true position, $P_i^{[k]}$ should be close to zero. Then, by Calculus techniques [32], we have

$$\begin{aligned} P_i^{[k+1]}-P_i^{[k]} &\approx dP_i \\ &= X'_i dx^e+Y'_i dy^e+Z'_i dz^e+\mathrm{E}'_i d\varepsilon^R \\ &\approx \begin{pmatrix} X'_i\left(x^{e[k+1]}-x^{e[k]}\right)+Y'_i\left(y^{e[k+1]}-y^{e[k]}\right) \\ +Z'_i\left(z^{e[k+1]}-z^{e[k]}\right)+\mathrm{E}'_i\left(\varepsilon^{R[k+1]}-\varepsilon^{R[k]}\right)\end{pmatrix}\end{aligned} \quad (3\text{-}25)$$

As we want the value of the residual function to become zero in the next iteration, we let $P_i^{[k+1]}=0$. Then, the above equation becomes:

$$0=\begin{pmatrix} P_i^{[k]} \\ +X'_i\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)\left(x^{e[k+1]}-x^{e[k]}\right) \\ +Y'_i\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)\left(y^{e[k+1]}-y^{e[k]}\right) \\ +Z'_i\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)\left(z^{e[k+1]}-z^{e[k]}\right) \\ +\mathrm{E}'_i\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)\left(\varepsilon^{R[k+1]}-\varepsilon^{R[k]}\right)\end{pmatrix} \quad (3\text{-}26)$$

As at the end of k-th iteration, the values of $P_i^{[k]},X'_i,Y'_i,Z'_i$, and E'_i are known for $i=1,\ 2,\ \ldots,\ m$. (3-26) specifies effectively a system of equations which we want to solve for $\left(x^{e[k+1]},y^{e[k+1]},z^{e[k+1]},\varepsilon^{R[k+1]}\right)$.

Given the above analysis, we are now ready to present the algorithm of the NR method:

Step 1: Determine an initial solution, e.g.,

$$\left(x^{e[k]},y^{e[k]},z^{e[k]},\varepsilon^{R[k]}\right)=(0,\ 0,\ 0,\ 0); \quad (3\text{-}27)$$

Step 2: Let k = 0;

Step 3: For $i=1,\ 2,\ \ldots,\ m$, compute the value of $P_i^{[k]}$ by (3-24);

Step 4: Solve (3-26) for $\left(x^{e[k+1]},y^{e[k+1]},z^{e[k+1]},\varepsilon^{R[k+1]}\right)$. In the case of $m>4$, the system of equations in (3-26) is over-determined. Hence, the *Ordinary Least Squares* (OLS for abbreviation) [16] method should be used to solve (3-26).

Step 5: If $P_i^{[k+1]}$ is small enough, stop. Otherwise continue.

Step 6: Let k = k + 1 and go to Step 3.

3.4.2. Remarks.

Application Assumptions. Some ideas of the NR algorithm may not be clear in the presentation of the algorithm. Especially, the algorithm makes several implicit approximations. We would like to reveal these approximations here in order to help the reader to understand and appreciate the algorithm.

First, (3-19) which is solved by the NR method, is an approximation of (3-17) which we really want to solve.

Second, in Step 4, a set of linear equations is solved. Here, the Taylor series [32] are effectively used to linearize the original quadratic equations. This helps to reduce the computational overhead.

Third, when $m > 4$, the system in (3-26) becomes over-determined. The OLS method is used to derive a solution. Note that the least square method is commonly utilized in this kind of situation in order to derive a solution that fits best with m given equations.

Validity of the Ordinary Least Square Method. Our reader may wonder if the OLS method is really valid here. In other words, under what kinds of conditions are the solutions derived with OLS reasonably accurate? In [11] [14] [15][24], extensive discussions have been given to argue that the OLS method is reasonable when it is used in practical positioning systems. Nevertheless, we state the major results from [11] that validate the approach taken in the algorithm.

The system of equations in (3-26) becomes over-determined when $m > 4$. In this case, generally no solution can satisfy all the equations in (3-26). That is, any solution will result in certain errors. As such, for a given solution $\left(x^{e[k]}, y^{e[k]}, z^{e[k]}, \varepsilon^{R[k]}\right)$, we may rewrite (3-26) as follows:

$$AX = B + V \tag{3-28}$$

where

$$A = \left.\begin{pmatrix} X_1^{'} & Y_1^{'} & Z_1^{'} & \mathrm{E}_1^{'} \\ X_2^{'} & Y_2^{'} & Z_2^{'} & \mathrm{E}_2^{'} \\ \cdots & \cdots & \cdots & \cdots \\ X_m^{'} & Y_m^{'} & Z_m^{'} & \mathrm{E}_m^{'} \end{pmatrix}\right|_{\left(x^{e[k]}, y^{e[k]}, z^{e[k]}, \varepsilon^{R[k]}\right)} \tag{3-29}$$

$$X = \begin{pmatrix} x^{e[k+1]} - x^{e[k]} \\ y^{e[k+1]} - y^{e[k]} \\ z^{e[k+1]} - z^{e[k]} \\ \varepsilon^{R[k+1]} - \varepsilon^{R[k]} \end{pmatrix}, \quad B = \begin{pmatrix} -P_1^{[k]} \\ -P_2^{[k]} \\ \cdots \\ -P_m^{[k]} \end{pmatrix} \tag{3-30}$$

where X'_i, Y'_i, Z'_i and E'_i are defined in (3-20), (3-21), (3-22) and (3-23), respectively. In (3-28), V is defined as follows:

$$V = \begin{pmatrix} v_1 \\ v_2 \\ \cdots \\ v_m \end{pmatrix} \tag{3-31}$$

where v_i is the error in the i-th equation.

Based on the above definitions, we now can state the optimality condition as follows: we say a solution is *optimal* if it minimizes the sum of squared errors. That is, an optimal solution minimizes

$$\sum_{i=1}^{m} (v_i)^2 \tag{3-32}$$

According to [16], the OLS method will generate the optimal solution if a solution satisfies the following conditions:

- $E(v_i) = 0$, (3-33)
- $\mathrm{var}(v_i) = \sigma^2 < \infty$, (3-34)
- $\mathrm{cov}(v_i, v_j) = 0, \ \ i \neq j$. (3-35)

where $i = 1, \ 2, \ \ldots, \ m$ and $m > 4$.

(3-33) means that the errors of all residuals in B have a zero-mean. (3-34) implies that the errors of all residuals in B have the same variance. (3-35) indicates that the errors of any two different residuals in B are uncorrelated.

Fortunately, with the parameter values taken from practical GPS systems [24], (3-26) meets all the conditions stated in (3-33), (3-34), and (3-35). Thus, the OLS method is valid in the NR algorithm.

4. Our Proposed Algorithms

4.1. The Basic Ideas

In this section, we discuss the major ideas utilized in our algorithms. Recall that in (3-17), the NR method treats ε_i^S as an error in ρ_i^e . In our algorithms, we also treat ε_i^S as an error in ρ_i^e . However, in other aspects, our algorithms are significantly different from the NR method. We discuss them in the following.

First, for ε^R , the receiver dependent error, the NR method treats it as an unknown variable and solves it during the iteration process. We intend to take a different approach. From the knowledge of clock bias in GPS receivers, we know that clock bias is predictable in practice (which will be discussed later) [12] [33]. *An idea in our algorithms is to use a prediction model to dynamically calibrate the clock.* That is, before we solve the equations, we will first estimate a value, say $\hat{\varepsilon}^R$, for ε^R by using a prediction model. Let the right side of (3-17) be ρ_i^E. We now have

$$\rho_i^E = \rho_i^e - \varepsilon^R \approx \rho_i^e - \hat{\varepsilon}^R \tag{4-1}$$

Thus, (3-17) can be written as follows:

$$\sqrt{(x_i - x^e)^2 + (y_i - y^e)^2 + (z_i - z^e)^2} \approx \rho_i^E \tag{4-2}$$

(4-2) is the system of equations we will try to solve. We will discuss in detail how to use the clock prediction model in a practical situation in Section 5.2.2. For the moment, our reader may assume that the value of $\hat{\varepsilon}^R$ and hence the value of ρ_i^E is given.

Our second idea is direct linearization. The NR method performs linearization during the iteration process by using Taylor series approximation. We will perform linearization directly on (4-2) by using algebraic techniques. By doing so, we can eliminate iterations and solve the equations directly, resulting in significant improvement in terms of computation time overhead. We will discuss this approach in Section 4.3.

The third idea is related to the least square method. In the NR method, the OLS method is used as the conditions in (3-33), (3-34), and (3-35) are met. Our case is different: (3-35) cannot be satisfied for the equations that are directly linearized. We nevertheless find that the *General Least Squares* (GLS for abbreviation) [16] method is valid for use. This will be discussed in Section 4.4.

4.2. Clock Bias Prediction

As we mentioned in Section 4.1, our method is based on a model of clock bias prediction. From (3-7), we know the true bias is Δt. As we do not know the exact value of Δt, a prediction model must be properly developed in order to obtain an approximate value of Δt.

Typically, in practice, a clock has a constant drift due to its stability on frequency. Then, $\Delta\hat{t}$, the estimation of Δt can be expressed as [15] [27] [30]:

$$\Delta t \approx \Delta\hat{t} = D + t_e r \tag{4-3}$$

where D is the offset at $t_e = 0$ and r is the clock drift. Therefore, we have $\hat{\varepsilon}^R$ as follows:

$$\hat{\varepsilon}^R = c\Delta\hat{t} \approx c(D + t_e r) \tag{4-4}$$

The question is how we can obtain values of D and r, so we can compute $\Delta\hat{t}$ or $\hat{\varepsilon}^R$ when needed. To do so, we must know the accurate standard time. In practice, two approaches have been taken: 1) To periodically acquire an accurate standard time from external time-keeping providers; 2) To use the clock bias calculated by the NR method as an approximation to $\Delta\hat{t}$ when external providers are not available [3] [10] [17] [33].

In Section 5, we will discuss practical examples of how to calculate D and r once the standard accurate time has been obtained or estimated.

Substituting (4-4) into (3-12), we have the revised formula for the pseudo-range as follows:

$$\rho_i^E = \rho_i^e - \hat{\varepsilon}^R = \rho_i^e - c(D + t_e r) \tag{4-5}$$

4.3. Direct Linearization Method

In this section, we will introduce our linearization method. Expanding the left side of (4-2), we have the following m equations: for $i = 1, \ldots, m$,

$$\begin{pmatrix} (x^e)^2 + (y^e)^2 + (z^e)^2 + x_i^2 + y_i^2 + z_i^2 \\ -2x_i x^e - 2y_i y^e - 2z_i z^e \end{pmatrix} = (\rho_i^E)^2 \tag{4-6}$$

where ρ_i^E is given in (4-1).

We can see in each equation that the coefficients of quadric items for x^e, y^e, and z^e are identical. Thus, these terms can be eliminated directly by subtraction. Now, if we subtract the first equation from the rest of equations, we have a system of $m-1$ linear equations:

$$\begin{pmatrix} (x_j - x_1)x^e \\ +(y_j - y_1)y^e \\ +(z_j - z_1)z^e \end{pmatrix} = \frac{1}{2}\begin{pmatrix} (x_j^2 - x_1^2) \\ +(y_j^2 - y_1^2) \\ +(z_j^2 - z_1^2) \\ -((\rho_j^E)^2 - (\rho_1^E)^2) \end{pmatrix} \tag{4-7}$$

where $j = 2, 3, \ldots, m$ and $m > 3$. Re-write (4-7) in a matrix form:

$$AX^e = D^e \tag{4-8}$$

where

$$A = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ \cdots & \cdots & \cdots \\ x_m - x_1 & y_m - y_1 & z_m - z_1 \end{bmatrix}, \tag{4-9}$$

$$X^e = \begin{bmatrix} x^e \\ y^e \\ z^e \end{bmatrix}, \tag{4-10}$$

and

$$D^e = \frac{1}{2}\begin{bmatrix} \begin{pmatrix} x_2{}^2 - x_1{}^2 + y_2{}^2 - y_1{}^2 + \\ z_2{}^2 - z_1{}^2 - (\rho_2^E)^2 + (\rho_1^E)^2 \end{pmatrix} \\ \begin{pmatrix} x_3{}^2 - x_1{}^2 + y_3{}^2 - y_1{}^2 + \\ z_3{}^2 - z_1{}^2 - (\rho_3^E)^2 + (\rho_1^E)^2 \end{pmatrix} \\ \dots \\ \begin{pmatrix} x_m{}^2 - x_1{}^2 + y_m{}^2 - y_1{}^2 + \\ z_m{}^2 - z_1{}^2 - (\rho_m^E)^2 + (\rho_1^E)^2 \end{pmatrix} \end{bmatrix} \tag{4-11}$$

Now, we need to solve (4-8). As the system is linear, we no longer need to use an iteration method.

4.4. Use of General Least Square Method

When $m > 3$, (4-8) becomes over-determined. Hence, a method like the OLS method may have to be used in order to derive a solution that fits best with the equations.

If we use the OLS method, the solution can be expressed as follows:

$$X^e = \left(A^T A\right)^{-1} A^T D^e \tag{4-12}$$

The question is whether (4-12) is optimal. The following theorem, unfortunately, indicates that the OLS method is not optimal here.

Theorem 4.1: When $m > 3$, for the system of equations in (4-8), conditions of the OLS method cannot be met, hence the OLS method is not optimal for solving (4-8).

Proof. We only need to show that condition (3-35) cannot be met. Let

$$\rho_i^E = \rho_i + \Delta\rho_i \tag{4-13}$$

where $\Delta\rho_j$ is the difference between the real distance and measured distance from S_j to the receiver. We also know from [24] that the following assumptions can be made:

$$E(\Delta\rho_j) = 0 \text{ and } \mathrm{var}(\Delta\rho_j) = \sigma^2 \tag{4-14}$$

$$\mathrm{cov}(\Delta\rho_i, \Delta\rho_j) = \sigma^2, \quad i \neq j \tag{4-15}$$

(4-14) means that $\Delta\rho_j$ is zero-mean and all having same variance σ^2 . (4-15) means that all pseudo-ranges are measured independently. Then, the errors in pseudo-ranges are uncorrelated from each other. Assume that the errors in D^e construct a column vector with $m-1$ elements:

$$\Delta\beta = \begin{bmatrix} \Delta\beta_1 \\ \Delta\beta_2 \\ \dots \\ \Delta\beta_{m-1} \end{bmatrix} \tag{4-16}$$

That is, the error $\Delta\beta_i$ in D_i^e is given by

$$\Delta\beta_i = \frac{1}{2}\begin{pmatrix} \begin{pmatrix} x_{i+1}{}^2 - x_1{}^2 + y_{i+1}{}^2 - y_1{}^2 + \\ z_{i+1}{}^2 - z_1{}^2 - (\rho_{i+1}^E)^2 + (\rho_1^E)^2 \end{pmatrix} \\ - \begin{pmatrix} x_{i+1}{}^2 - x_1{}^2 + y_{i+1}{}^2 - y_1{}^2 + \\ z_{i+1}{}^2 - z_1{}^2 - (\rho_{i+1})^2 + (\rho_1)^2 \end{pmatrix} \end{pmatrix} \tag{4-17}$$

where $i = 1,\ 2,\ \dots,\ m-1$.

Hence,

$$\Delta\beta = \frac{1}{2}\begin{bmatrix} (\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_2 + 2\rho_2)\Delta\rho_2 \\ (\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_3 + 2\rho_3)\Delta\rho_3 \\ \dots \\ (\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_m + 2\rho_m)\Delta\rho_m \end{bmatrix} \tag{4-18}$$

Next, we will calculate the covariance for any two different $\Delta\beta_i$. We need to calculate the expected value for $\Delta\beta_i$ first, we have

$$\begin{aligned}
&E(\Delta\beta_i) \\
&= \frac{1}{2}E((\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_{i+1} + 2\rho_{i+1})\Delta\rho_{i+1}) \\
&= \frac{1}{2}E\left(\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1 - \Delta\rho_{i+1}{}^2 - 2\rho_{i+1}\Delta\rho_{i+1}\right) \\
&= \frac{1}{2}\left(E\left(\Delta\rho_1{}^2\right) + 2\rho_1 E(\Delta\rho_1) - E\left(\Delta\rho_{i+1}{}^2\right) - 2\rho_{i+1}E(\Delta\rho_{i+1})\right) \\
&= \frac{1}{2}\left(E\left(\Delta\rho_1{}^2\right) - E\left(\Delta\rho_{i+1}{}^2\right)\right) \\
&= \frac{1}{2}\left(\mathrm{var}(\Delta\rho_1) + E^2(\Delta\rho_1) - \mathrm{var}(\Delta\rho_{i+1}) - E^2(\Delta\rho_{i+1})\right) \\
&= \frac{1}{2}\left(\mathrm{var}(\Delta\rho_1) - \mathrm{var}(\Delta\rho_{i+1})\right) \\
&= \frac{1}{2}\left(\sigma^2 - \sigma^2\right) = 0
\end{aligned}$$

That is

$$E(\Delta\beta_i) = 0 \tag{4-19}$$

Then, we have

$$\begin{aligned}
&\mathrm{cov}(\Delta\beta_i, \Delta\beta_j) \\
&= E\left((\Delta\beta_i - E(\Delta\beta_i))(\Delta\beta_j - E(\Delta\beta_j))\right) \\
&= E(\Delta\beta_i\Delta\beta_j) \\
&= \frac{1}{4}E\begin{pmatrix} ((\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_{i+1} + 2\rho_{i+1})\Delta\rho_{i+1}) \\ ((\Delta\rho_1 + 2\rho_1)\Delta\rho_1 - (\Delta\rho_{j+1} + 2\rho_{j+1})\Delta\rho_{j+1}) \end{pmatrix} \\
&= \frac{1}{4}E\begin{pmatrix} ((\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1) - (\Delta\rho_{i+1}{}^2 + 2\rho_{i+1}\Delta\rho_{i+1})) \\ ((\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1) - (\Delta\rho_{j+1}{}^2 + 2\rho_{j+1}\Delta\rho_{j+1})) \end{pmatrix}
\end{aligned}$$

$$= \frac{1}{4}\begin{pmatrix} E\left((\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1)(\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1)\right) \\ - E\left((\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1)(\Delta\rho_{i+1}{}^2 + 2\rho_{i+1}\Delta\rho_{i+1})\right) \\ - E\left((\Delta\rho_{j+1}{}^2 + 2\rho_{j+1}\Delta\rho_{j+1})(\Delta\rho_1{}^2 + 2\rho_1\Delta\rho_1)\right) \\ + E\left((\Delta\rho_{j+1}{}^2 + 2\rho_{j+1}\Delta\rho_{j+1})(\Delta\rho_{i+1}{}^2 + 2\rho_{i+1}\Delta\rho_{i+1})\right) \end{pmatrix}$$

$$= \frac{1}{4}\begin{pmatrix} E\left(\Delta\rho_1{}^2\Delta\rho_1{}^2\right) + 4\rho_1{}^2 E\left(\Delta\rho_1{}^2\right) \\ - E\left(\Delta\rho_1{}^2\Delta\rho_{i+1}{}^2\right) - E\left(\Delta\rho_{j+1}{}^2\Delta\rho_1{}^2\right) \\ + E\left(\Delta\rho_{j+1}{}^2\Delta\rho_{i+1}{}^2\right) \end{pmatrix}$$

$$= \rho_1{}^2\sigma^2$$

where $i = 1, 2, ..., m-1$ and $j = 1, 2, ..., m-1$.

That is

$$\mathrm{cov}\left(\Delta\beta_i, \Delta\beta_j\right) = \rho_1{}^2\sigma^2 \neq 0 \quad (4\text{-}20)$$

Therefore, Theorem 4.1 is proven. ■

Thus, while the OLS method may still be used, given Theorem 4.1, it would be better if we can identify an optimal method to solve (4-8). Let us consider the General Least Squares (GLS) method. According to [16], the GLS method derives the solution by the following formula:

$$X^e = \left(A^T M^{-1} A\right)^{-1} A^T M^{-1} D^e \quad (4\text{-}21)$$

where M is the covariance matrix of $\Delta\beta$:

$$M = \mathrm{cov}(\Delta\beta) = E\left(\Delta\beta\Delta\beta^T\right) \quad (4\text{-}22)$$

The optimality requirement of the GLS method is less than the OLS method. According to [16], (4-21) is optimal if the following conditions hold:

- $E(\Delta\beta_i) = 0$, (4-23)
- $\mathrm{cov}(\Delta\beta) = \sigma^2\Omega$, (4-24)

where Ω is a positive definite matrix but may not be an identity matrix. (4-23) means all elements in $\Delta\beta$ are zero-mean; (4-24) means that different elements in $\Delta\beta$ now can be correlated. *Optimal* here means that the sum of squared errors, i.e. $\Delta\beta_i$, is minimal.

We have the following theorem to show that the solution given by (4-21) is optimal.

Theorem 4.2: When $m > 3$, for the system of equations in (4-8), conditions of the GLS method can be met. Hence, the GLS method is optimal for solving (4-8).

Proof. To prove the theorem, we need to show that conditions in (4-23) and (4-24) are met. First, from (4-19), we know $E(\Delta\beta_i) = 0$. Thus, the condition in (4-23) is satisfied.

Now, let us calculate the covariance matrix for $\Delta\beta$. From [21], we have the following:

$$\mathrm{cov}(\Delta\beta) = \sigma^2\Psi \quad (4\text{-}25)$$

where

$$\Psi = \begin{bmatrix} \rho_2{}^2 + \rho_1{}^2 & \rho_1{}^2 & \cdots & \rho_1{}^2 \\ \rho_1{}^2 & \rho_3{}^2 + \rho_1{}^2 & \cdots & \rho_1{}^2 \\ \cdots & \cdots & \cdots & \cdots \\ \rho_1{}^2 & \rho_1{}^2 & \cdots & \rho_m{}^2 + \rho_1{}^2 \end{bmatrix} \quad (4\text{-}26)$$

By (4-26), we see that Ψ is not an identity matrix but a positive definite matrix, so the condition in (4-24) is also satisfied. Theorem 4.2 is proven. ■

4.5. Our Algorithms

Based on the discussions above, we propose two algorithms:

- *Algorithm DLO*: uses the direct linearization with the OLS method, and
- *Algorithm DLG*: uses the direct linearization with the GLS method.

The pseudo codes of the algorithms are given below.

Algorithm DLO

Step 1. Calculate $\hat{\varepsilon}^R$;

Step 2. Calculate ρ_i^E by (4-1) and substitute it into (4-10);

Step 3. Calculate X^e by the *ordinary least squares* method, i.e.,

$$X^e = \left(A^T A\right)^{-1} A^T D^e \quad (4\text{-}12)$$

Algorithm DLG

Step 1. Calculate $\hat{\varepsilon}^R$;

Step 2. Calculate ρ_i^E by (4-1);

Step 3. Calculate covariance matrix M of D^e.

Step 4. Calculate X^e by the *generalized least squares* method, i.e.,

$$X^e = \left(A^T M^{-1} A\right)^{-1} A^T M^{-1} D^e \quad (4\text{-}21)$$

In Section 5, we will compare the performance of the algorithms with the NR method.

5. Performance Evaluation

5.1. Performance Metrics

We will first define absolute performance metrics. We will then define performance metrics related to that of the NR method in order to carry out an effective comparison.

Table 5.1. Data Set Specifications

No.	Site ID	ECEF Coordinates (X, Y, Z)(m)	Date of Collection	Clock Correction Type
1	SRZN	(3623420.032,-5214015.434, 602359.096)	2009/08/12	Steering
2	YYR1	(1885341.558, -3321428.098, 5091171.168)	2009/10/23	Steering
3	FAI1	(-2304740.630, -1448716.218, 5748842.956)	2009/10/29	Steering
4	KYCP	(411598.861, -5060514.896, 3847795.506)	2009/10/10	Threshold

Let a GPS system use a particular location algorithm O where O can be NR, DLO, or DLG, as discussed in Sections 3 and 4. If the true position of the GPS receiver is (x, y, z) and the estimated position is (x^e, y^e, z^e), the absolute error of the GPS system is then given by

$$d_o = \sqrt{(x^e - x)^2 + (y^e - y)^2 + (z^e - z)^2} \qquad (5\text{-}1)$$

Accuracy Rate for location algorithm O is defined as follows:

$$\eta = \frac{d_O}{d_{NR}} \times 100\% \qquad (5\text{-}2)$$

where subscribe O can be either DLO or DLG. If $\eta > 100\%$, it means the accuracy of location algorithm O is worse than NR; otherwise it is better.

Let *execution time for executing algorithm O be* τ_O where O can be NR, DLO, or DLG as discussed in Sections 3 and 4, respectively. Execution Time *Rate for algorithm* O *is given by*

$$\theta = \frac{\tau_O}{\tau_{NR}} \times 100\% \qquad (5\text{-}3)$$

where subscribe O can be either DLO or DLG. If $\theta < 100\%$, it means method O is better than NR method; otherwise method O is worse.

5.2. Experiment Settings

5.2.1. Data Sets

We downloaded four data sets randomly selected from land observation stations [8]. We chose four different locations and obtained four data sets. Table 5.1 provides the specifications of these data. All measurements are based on the L1 signal. Each data set contains 24-hour observational data, which contains 86,400 data items. That is, for every second, all available satellites' coordinates and pseudo-ranges are contained in one data item. Generally each item contains data for 8 to 12 satellites.

Our objective here is to use each of these three methods, namely NR, DLO, and DLG, to compute an estimation for the location of the GPS observation station.

5.2.2. Clock Bias Prediction on Data Sets

When method DLO or DLG is used, we will use the method in (4-4) to calibrate the clock.

It is our understanding that observation stations use different approaches, say, *clock steering approach* or *threshold approach,* to calibrate their clocks [15][24]. Our model discussed in (4-3) covers both approaches. We briefly describe these methods here. For details, please refer to [15] [24].

With the steering approach, the system manages to control $t_e r$ within a small range of standard time. With the threshold approach, $t_e r$ will change as the passage of time. Whenever the clock error reaches a pre-set threshold, the clock will be adjusted. After knowing the behavior of clock bias in the data, we can use following method calculate D and r :

For the system with the steering approach, D is calculated only once at the initialization time (i.e., for the systems with data sets 1, 2 and 3). For the system with the threshold approach, D is calculated whenever clock bias is reset. In either case, we use the NR method to derive ε^R and then compute D by the following formula:

$$D \approx \varepsilon^R / c \qquad (5\text{-}4)$$

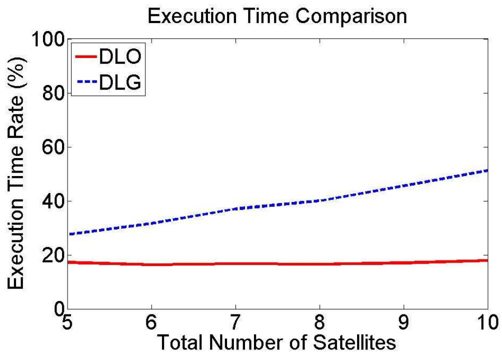
For clock drift r , a small set of data items at the initialization time is used to compute it.

5.2.3. Computation Environment

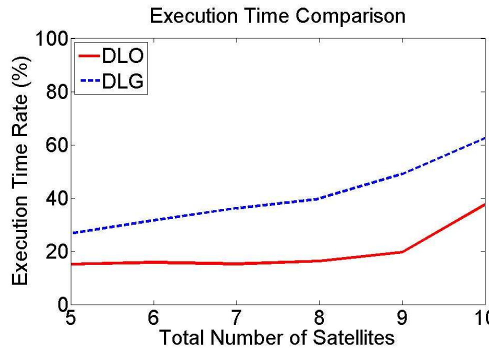
When executing the location algorithms with a computer system with the following configurations: It has an AMD Dual Processor with 2GHz and 2MB of Cache, 2GB Physical Memory, and 150GB Disk. The operating system is Linux and *C* is used as the programming language.

5.3. Observations on Execution Time Rates

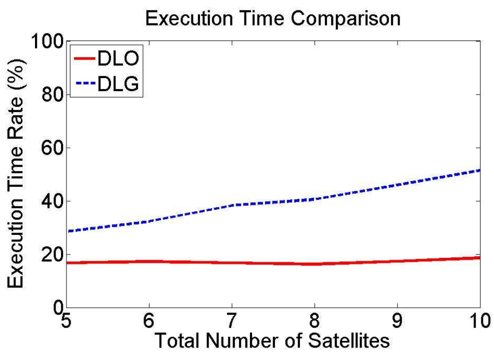
From Fig. 5.1, we can see that execution times of the DLO and DLG methods are significantly less than that of the NR method. The average execution time of Algorithm DLO only takes typically less than 20% of that of the NR method, indicating a significant improvement.



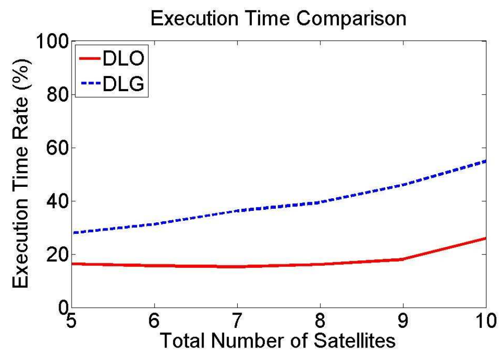
(a) Data Set 1



(b) Data Set 2

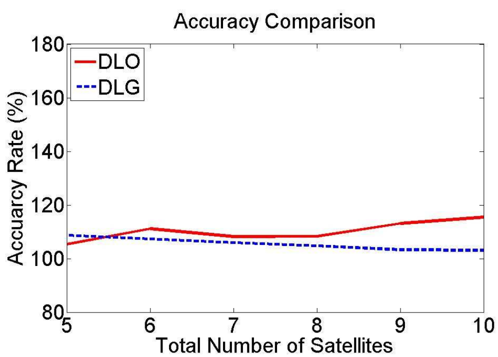


(c) Data Set 3

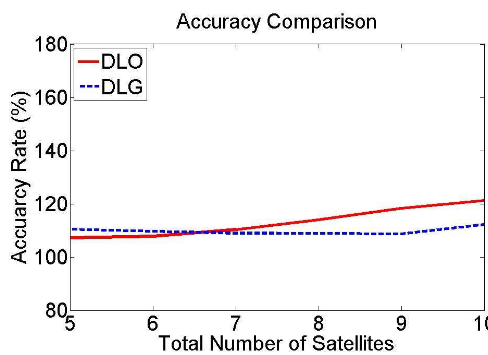


(d) Data Set 4

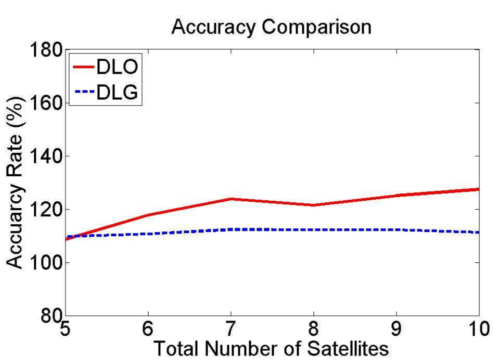
Figure 5.1 Execution Time Comparisons



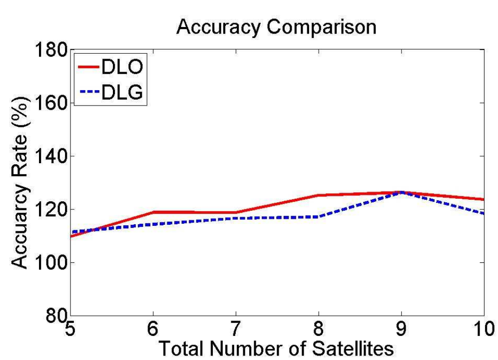
(a) Data Set 1



(b) Data Set 2



(c) Data Set 3



(d) Data Set 4

Figure 5.2 Accuracy Comparisons

For the algorithm DLG, as the number of satellites increases, the execution time rate increases. Nevertheless, even if the number of satellites is 10, the execution time rate is mostly about 50%, indicating still very good performance improvement in comparison with the NR method.

5.4. Observations on Accuracy Rates

From Fig 5.2, we can see that accuracy of algorithms DLO and DLG is close to the NR method. For DLG algorithm, it is only slightly different from NR (accuracy rate is around 110%). Regardless of whether there are more or less satellites, accuracy rate of DLG remains almost constant. For algorithm DLO, its accuracy becomes worse when the number of satellites increases. When there are 10 satellites, the accuracy rate decreases to around 120%.

The reason why algorithm DLO's accuracy becomes worse when using more satellites is that, when more satellites are used, more errors are introduced to equation system in (4-12). This will decrease the accuracy of algorithm DLO. However, for the DLG algorithm, the use of covariance matrix in (4-25) will eliminate the effect of these errors, so its accuracy rate will keep as a constant.

6. Final Remarks

In this paper, we have discussed and analyzed a new algorithm for the GPS positioning problem. Our approach uses a direct linearization method. In comparison with the traditional NR method, our proposed algorithm can achieve similar performance in terms of positioning accuracy while taking much less execution time. Typically, our new methods take about one fifth of the computation time required by the traditional NR method.

Our investigation is preliminary and many extensions are possible. For example, the accuracy can be further improved if we can identify a "good" satellite to be used as the base to construct the linear system. In the algorithm we propose in this paper, this satellite is randomly chosen. Another extension is to consider better clock bias models so the clock prediction can be further improved along with the accuracy of the algorithm. The third extension is to optimize the matrix operations in the context of our problem so the computation time may be further reduced.

Acknowledgements

We thank the anonymous reviewers for their invaluable feedback. This work was supported in part by the US National Science Foundation (NSF) under grants 0963973 and 0963979 and by the University of Macau, and Macao Science and Technology Development Fund. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency and institution.

References

[1] J. S. Abel and J. W. Chaffee, "Existence and uniqueness of GPS solutions," *IEEE Trans. on Aerospace & Electronic Systems*, 27(6), 1991, pp. 952-956.

[2] S. Bancroft, "An algebraic solution of the GPS equations," *IEEE Trans. on Aerospace & Electronic Systems*, 21(7), 1986, pp. 56-59.

[3] S. Bednarz, *Adaptive Modeling of Receiver Clock for Integrity Monitoring During Precision Approaches*, Master's Thesis, Dept. of Aeronautics and Astronautics, MIT, 2004.

[4] I. Biton, M. Koifman and I. Y. Bar-Itzhack, "Improved Direct Solution of the Global Positioning System Equation," *Journal of Guidance, Control, and Dynamics*, 1998, pp. 1313 - 1320.

[5] A. Beck, P. Stoica and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Trans. on Signal Processing*, 56(5), 2008, pp. 1770-1778.

[6] J. Chaffee and J. Abel, "On the Exact Solutions of Pseudo-range Equations," *IEEE Trans. on Aerospace and Electronic Systems*, 30(4), 1994, pp. 1021-1030.

[7] K. W. Cheung, W.-K. Ma and H. C. So, "Accurate approximation algorithm for toa-based maximum likelihood mobile location using semidefinite programming," *IEEE International Conf. on Acoustics, Speech, and Signal Processing*, 2004, pp. II 145- II 148.

[8] CORS, Continuously Operating Reference Stations, National Geodetic Survey, http://www.ngs.noaa.gov/CO RS/cors-data.html.

[9] K. Cheung, H. So, W.-K. Ma and Y. Chan, "Least squares algorithms for time-of-arrival-based mobile location," *IEEE Trans. on Signal Processing*, 52, 2004, pp. 1121-1130.

[10] P.H. Dana, "Global Positioning System (GPS) Time Dissemination for Real-Time Applications," *Journal of Real-Time Systems*, 12(1), 1997, pp. 9-40.

[11] D. J. Dailey and B.M. Bell, "A method for GPS positioning," *IEEE Trans. on Aerospace & Electronic Systems*, 32(3), 1996, pp. 1148-1154.

[12] E. A. Marques Filho, H. K. Kuga, R. V. F. Lopes, "Real time estimation of GPS receiver clock offset by the Kalman filter", *Personal Communication*, 2003.

[13] E. W. Grafarend and J. Shan, "GPS solutions: closed forms, critical and special configurations of P4P", *GPS Solutions*, 5(3), 2002, pp. 29-41.

[14] M. S. Grewal, L. R. Weill and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, John Wiley & Sons, 2001.

[15] E. D. Kaplan (ed.), *Understanding GPS: principles and applications*, Norwood, Artech House, 1996.

[16] T. Kariya and H. Kurata, *Generalized Least Squares*, Chichester, UK: John Wiley & Sons, Ltd., 2004.

[17] P. A. Kline, *Atomic Clock Augmentation for Receivers Using the Global Positioning System*, Ph. D Dissertation, Dept. of Electrical Engineering, Virginia Polytechnic Institute and State University, 1997

[18] L. O. Krause, "A Direct Solution to GPS Type Navigation Equations," *IEEE Trans. on Aerospace and Electronic Systems*, 23(2), 1987, pp. 223-232.

[19] J. Hoshen, "The GPS Equations and the Problem of Apollonius," *IEEE Trans. on Aerospace and Electronic Systems*, 32(3), 1996, pp. 1116-1124.

[20] R. P. Langley, "The Mathematics of GPS," *GPS World*, 1991, pp 45-50.

[21] W. Li, L. Deng, S. Yang, Z. Xu and W. Zhao, *A Direct Solution for GPS positioning*, Technical Report, FST, University of Macau, 2009.

[22] J. L. Leva, "An alternative closed-form solution to the GPS pseudorange equations," *IEEE Trans. on Aerospace & Electronic Systems*, 32(4), 1996, pp. 1430-1439.

[23] J. B. Lundberg, "Alternative algorithms for the GPS static positioning solution," *Applied Mathematics and Computation*, 119(1), 2001, pp. 21-34.

[24] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance*, Ganga-Jamuna Press, Lincoln, Mass., 2001.

[25] P. S. Noe, K. A. Myers and T. K. Wu, "A navigation algorithm for low cost GPS receiver," *Journal of the Institute of Navigation*, 25 (2), 1978, pp. 258-264.

[26] S. Nardi and M. Pachter, "GPS estimation algorithm using stochastic modeling", *Proceedings of 37th Conference on Decision & Control*, 1998, pp. 4498-4502.

[27] P. Misra, "The Role of the Clock in a GPS Receiver," *GPS World*, 1996, pp. 60-66.

[28] M. Pachter and T. Q. Nguyen, "An Efficient GPS Position Determination Algorithm," *Journal of the Institute of Navigation*, 50(2), 2003, pp. 131-141.

[29] B. Parkinson (ed.) and J. Spilker (ed.), *Global positioning system: theory and applications Volume I*, Cambridge, Charles Stark Draper Laboratory, Inc., 1996.

[30] M. A. Sturza, "GPS navigation using three satellites and a precise clock," *Navigation: Journal of the Institute of Navigation*, 30(2), 1983, pp. 146-156.

[31] A. Sayed, A. Tarighat and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, 22(4), 2005, pp. 24-40.

[32] G. B. Thomas, R. L. Finney, *Calculus and Analytic Geometry (9th ed.)*, Addison Wesley, 1996.

[33] C. Thomas, "Real-Time Restitution of GPS time through a Kalman Estimation," *Metrologia*, 29, 1992, pp. 397-414.

[34] Z. Zhang, *Impact of Rubidium Clock Aiding on GPS Augmented Vehicular Navigation*, M.Sc. Thesis, Dept. of Geomatics Engineering, The University of Calgary, 1997.

[35] N. Zhang and W. Zhao, "Distributed Privacy Preserving Information Sharing," in the *Proceedings of International Conference on Very Large Data Bases (VLDB)*, 2005, pp. 889-900.

[36] W. Zhao and J. Stankovic, "Performance Analysis of FCFS and Improved FCFS Scheduling Algorithms for Dynamic Real-Time Computer Systems", *Proceedings of IEEE Real-Time Systems Symposium*, 1989, pp. 156-165.