Development of a Sequential Algorithm for a GNSS-Based Multi-Sensor Vehicle Navigation System

Chang Wan Jeon, Gyu-In Jee, and Gérard Lachapelle

Abstract: RAIM techniques based on TLS have rarely been addressed because TLS requires a great number of computations. In this paper, the particular form of the observation matrix $\mathbf{H}$, is exploited so as to develop a new TLS-based sequential algorithm to identify an errant satellite. The algorithm allows us to enjoy the advantages of TLS with less computational burden. The proposed algorithm is verified through a numerical simulation.

Keywords: GPS, RAIM, sequential algorithm, Total Least Squares (TLS).

1. INTRODUCTION

Since the mid 1980s, receiver autonomous integrity monitoring (RAIM) has received a great deal of attention. This is because the integrity information provided via a navigation message may not be timely enough in some applications. Nowadays, extensive researches on this topic have been performed under the name of RAIM, FDI (failure detection and isolation), and FDE (failure detection and exclusion) [1,2].

One of the major methods of GPS positioning and integrity monitoring is to formulate a linear measurement model and solve it using the least squares (LS) method to obtain an appropriate position and clock bias [3,4]. Particularly in the case of integrity monitoring, many algorithms have formed n subsets of n-1 satellites by sequentially deleting one satellite not previously excluded and have calculated test statistics for each satellite subset using least squares estimates [5-7].

Recently Juang [8] reformulated a linear measurement model and proposed a positioning and integrity monitoring scheme based on total least squares (TLS) instead of least squares. He took advantage of TLS at the expense of more numerical computations by proposing a new integrity monitoring metric. The new metric employs a minimum singular value from SVD that is almost indispensable to solving the TLS problem. Juang's scheme is suitable for failure detection.

In this paper, a new TLS-based sequential algorithm focused on failure isolation, even though it can be used for failure detection, is proposed. This algorithm utilizes the fact that a linear matrix equation has a precisely known column. The algorithm takes a sequential form so as to reduce the amount of computations necessary. It makes use of previous results without repeating the entire process. Therefore one can enjoy, with less computational burden, the advantages for integrity monitoring provided by Juang who employed TLS as a tool for positioning and integrity monitoring

2. TECHNICAL BACKGROUND

This section describes the linear measurement model and mixed LS-TLS problem for discussion in the remainder of the paper.

2.1. Linear measurement model

A linear model is generally employed for proper positioning and integrity monitoring. In [8], the linear measurement model was reinvestigated considering errors in observation matrix H. In this model, error due to a failed satellite is included in observation matrix H. Therefore, the observation matrix H is not exactly known any longer. Naturally, TLS is employed to solve this problem. In this paper, the linear model in [8] will be used. Therefore, a brief description of the model is given.

Suppose n satellites are visible. The measurement model is

Manuscript received January 29, 2003; revised March 31, 2004; accepted April 26, 2004. Recommended by Editorial Board member Jin Young Choi under the direction of Editor Jin Bae Park. This work was supported by the postdoctoral fellowships program from the Korea Science and Engineering Foundation (KOSEF).

Chang-Wan Jeon is with the Division of Information Technology Engineering, Soonchunhyang University, San 53-1 Eupnae-ri, Shinchang-myun, Asan-si, Choongnam 336-745, Korea (e-mail: jeoncw@sch.ac.kr).

Gyu-In Jee is with the Department of Electronics Engineering, Konkuk University, Hwayang-dong, Gwangjin-gu, Seoul 143-701 (e-mail: gijee@kkucc.konkuk.ac.kr).

Gérard Lachapelle is with the Department of Geomatics Engineering, University of Calgary, 2500 University Drive N.W., Calgary, Alberta, Canada T2N 1N4 (e-mail: lachapel@geomatics.ucalgary.ca).

$$\rho^i = \left\| \mathbf{u} - \mathbf{s}^i - \Delta\mathbf{s}^i \right\| + c + e^i , \tag{1}$$

where ρ^i is the pseudo-range measurement with respect to the i-th GPS satellite, $\mathbf{u}$ is the user's position, c is clock offset, $\mathbf{s}^i$ is the broadcast position of the i-th GPS satellite, $\Delta\mathbf{s}^i$ is the difference between the broadcast position and true position of the i-th GPS satellite, and e^i accounts for the additional errors. e^i is treated as zero mean noise. Both the pseudo-range ρ^i and the broadcast position $\mathbf{s}^i$ are subject to errors due to ephemeris errors, SA effects if it exists, environment effects, satellite failure, interferences, noises, and so on. Let

$$\mathbf{u} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \ \mathbf{s}^i = \begin{bmatrix} x^i \\ y^i \\ z^i \end{bmatrix}, \text{and } \Delta\mathbf{s}^i = \begin{bmatrix} \Delta x^i \\ \Delta y^i \\ \Delta z^i \end{bmatrix}. \tag{2}$$

Suppose that the linearization point is at

$$\mathbf{u} = \mathbf{u}_0 = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \text{ and } c = c_0 \tag{3}$$

then the estimate of the psudo-range measurement is given by

$$\begin{aligned} \rho_0^i &= \left\| \mathbf{u}_0 - \mathbf{s}^i - \Delta\mathbf{s}^i \right\| + c_0 \\ &= \sqrt{\left(x^i + \Delta x^i - x_0\right)^2 + \left(y^i + \Delta y^i - y_0\right)^2 + \left(z^i + \Delta z^i - z_0\right)^2} + c_0 \end{aligned} \tag{4}$$

Define

$$\mathbf{r} = \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix}, \ \delta = -c + c_0 \text{, and } \mathbf{p} = \begin{bmatrix} \mathbf{r} \\ \delta \end{bmatrix}. \tag{5}$$

Then, the linearized matrix equation of (1) with respect to n observable satellites becomes

$$\mathbf{Hp} = \mathbf{q} + \mathbf{e} , \tag{6}$$

where

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & 1 \\ h_{21} & h_{22} & h_{23} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ h_{n1} & h_{n2} & h_{n3} & 1 \end{bmatrix}, \ \mathbf{q} = \begin{bmatrix} \rho_0^1 - \rho^1 \\ \rho_0^2 - \rho^2 \\ \vdots \\ \rho_0^n - \rho^n \end{bmatrix}, \ \mathbf{e} = \begin{bmatrix} e^1 \\ e^2 \\ \vdots \\ e^n \end{bmatrix},$$

and

$$h_{i1} = \frac{x^i + \Delta x^i - x_0}{\sqrt{\left(x^i + \Delta x^i - x_0\right)^2 + \left(y^i + \Delta y^i - y_0\right)^2 + \left(z^i + \Delta z^i - z_0\right)^2}}$$

$$h_{i2} = \frac{y^i + \Delta y^i - y_0}{\sqrt{\left(x^i + \Delta x^i - x_0\right)^2 + \left(y^i + \Delta y^i - y_0\right)^2 + \left(z^i + \Delta z^i - z_0\right)^2}}$$

$$h_{i3} = \frac{z^i + \Delta z^i - z_0}{\sqrt{\left(x^i + \Delta x^i - x_0\right)^2 + \left(y^i + \Delta y^i - y_0\right)^2 + \left(z^i + \Delta z^i - z_0\right)^2}}.$$

Note that the last column of the H matrix is precisely identified. Therefore solving the linearized matrix equation is a mixed LS-TLS problem described in the next section.

2.2. Mixed LS-TLS problem

Let $\mathbf{b} = \mathbf{q} + \mathbf{e}$ for simplicity. Then (6) becomes a well-known linear matrix equation,

$$\mathbf{Hp} = \mathbf{b} . \tag{7}$$

In the classical LS approach, all elements of $\mathbf{H}$ are assumed to be free of error; hence, all errors are confined to the observation vector $\mathbf{b}$. This assumption, however, is frequently unrealistic in various applications. The TLS is one method of fitting that is appropriate when there are errors in both the observation vector $\mathbf{b}$ and the data matrix $\mathbf{H}$. When some of the columns of the data matrix $\mathbf{H}$ are free of errors, like the case considered in this paper, we refer to it as a mixed LS-TLS problem [9]. In this case, we can solve the mixed problem by solving the LS and TLS problem separately with a proper batch algorithm in [9]. Unfortunately, the batch process demands a great deal of computational effort. It is inappropriate to apply GPS integrity monitoring because of computational burden. An approach to reduce the computational effort is to construct an algorithm in a recursive (or sequential) form. In this paper, a sequential algorithm, which has many advantages over the batch algorithm, is proposed for effectively explaining the mixed LS-TLS dilemma discussed in the previous section for GPS integrity monitoring. It starts from the batch algorithm. Therefore, the existing batch algorithm is reviewed in this section.

We can permute the order of columns in $\mathbf{H}$ with a proper permutation matrix and obtain the following equation without loss of the generality,

$$\mathbf{Ax} = \mathbf{b} , \tag{8}$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & h_{11} & h_{12} & h_{13} \\ 1 & h_{21} & h_{22} & h_{23} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & h_{n1} & h_{n2} & h_{n3} \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} \delta \\ \mathbf{r} \end{bmatrix}. \tag{9}$$

This trick is for convenience only. Let a matrix

$\mathbf{A}=\left[\mathbf{A}_1;\mathbf{A}_2\right]$ be given whose first p columns $\mathbf{A}_1$ have no error and have full column rank. Suppose the matrix have p exactly known columns to generalize the discussion, although it has only one exactly known column in this case. Then, the algorithm is as follows. Perform p Householder transformations $\mathbf{Q}$ on the matrix $[\mathbf{A};\mathbf{b}]$ so that

$$\mathbf{Q}^T\left[\mathbf{A}_1;\mathbf{A}_2;\mathbf{b}\right]=\begin{bmatrix}\mathbf{R}_{11} & \mathbf{R}_{12} & \mathbf{y}_1\\ 0 & \mathbf{R}_{22} & \mathbf{y}_2\end{bmatrix}, \tag{10}$$

where $\mathbf{R}_{11}$ is a $p\times p$ upper triangular matrix. (scalar in this case). Then, compute the TLS solution $\hat{\mathbf{x}}_2$ of $\mathbf{R}_{22}\mathbf{x}_2=\mathbf{y}_2$ by the SVD. $\hat{\mathbf{x}}_2$ yields the last *n-p* components of the solution vector $\hat{\mathbf{x}}$. To find the first p components $\hat{\mathbf{x}}_1$ of the solution vector, solve

$$\mathbf{R}_{11}\mathbf{x}_1=\mathbf{y}_1-\mathbf{R}_{12}\mathbf{x}_2. \tag{11}$$

This is simply the LS solution obtained by projecting the reduced observation vector $\mathbf{b}-\mathbf{A}_2\hat{\mathbf{x}}_2$ into the space $R(\mathbf{A}_1)$ generated by the known columns of $\mathbf{A}$.

Note that it requires a great number of operations if we repeat the whole process for n subsets of n-1 satellites to identify an errant satellite.

3. DERIVATION OF A SEQUENTIAL ALGORITHM TO IDENTIFY A FAILED SATELLITE

Suppose the *k*-th satellite is deleted. Let $\mathbf{A}_k=\left[\mathbf{A}_{k,1}\quad \mathbf{A}_{k,2}\right]$ be the matrix formed by deleting the *k*-th row of $\mathbf{A}$. $\mathbf{A}_{k,1}$ is assumed to be exactly known. Let $\mathbf{b}_k$ be the observation vector formed by deleting the *k*-th element of $\mathbf{b}$. Suppose we obtained the Householder transformation matrix $\mathbf{Q}_k$ on the matrix $\left[\mathbf{A}_k;\mathbf{b}_k\right]$ so that

$$\mathbf{Q}_k^T\left[\mathbf{A}_{k,1};\mathbf{A}_{k,2};\mathbf{b}_k\right]=\left[\mathbf{R}_k;\,\mathbf{B}_k\right], \tag{12}$$

where $\mathbf{R}_k=\mathbf{Q}_k^T\mathbf{A}_{k,1}\in\Re^{(n-1)\times p}$ is an upper triangular matrix and $\mathbf{B}_k=\mathbf{Q}_k^T\left[\mathbf{A}_{k,2};\,\mathbf{b}_k\right]$. At the next step, (*k*+1)-th satellite is deleted and k-th satellite is included. The algorithm takes two steps, deleting the (*k*+1)-th satellite ((*k*+1)-th row) and adding the previously deleted k-th satellite (*k*-th row). When the previously deleted satellite is included, the data and observation matrices are given as follows:

$$\mathbf{A}_{k+1}^+=\begin{bmatrix}\mathbf{A}_k\\ \boldsymbol{\alpha}_k^T\end{bmatrix} \text{ and } \mathbf{b}_{k+1}^+=\begin{bmatrix}\mathbf{b}_k\\ \beta_k\end{bmatrix}, \tag{13}$$

where $\boldsymbol{\alpha}_k$ and β_k are the previously deleted *k*-th row. Then, the (*k*+1)-th satellite (the first row of $\mathbf{A}_k$ and $\mathbf{b}_k$) should be deleted. Denote the matrix and the vector obtained by deleting the first column from $\mathbf{A}_k$ and $\mathbf{b}_k$ by $\bar{\mathbf{A}}_k=\left[\bar{\mathbf{A}}_{k,1};\,\bar{\mathbf{A}}_{k,2}\right]$ and $\bar{\mathbf{b}}_k$, respectively.

As the first step, consider the (*k*+1)-th satellite deleting case. It is claimed to compute the QR factorization of $\bar{\mathbf{A}}_{k,1}$, *i.e.*,

$$\bar{\mathbf{A}}_{k,1}=\bar{\mathbf{Q}}_k\bar{\mathbf{R}}_k \tag{14}$$

for the sequential algorithm. It is also required to compute the matrix $\bar{\mathbf{B}}_k$

$$\bar{\mathbf{B}}_k=\bar{\mathbf{Q}}_k^T\left[\bar{\mathbf{A}}_{k,2};\,\bar{\mathbf{b}}_k\right]. \tag{15}$$

For sequential processing of the above operations, we must update the $\bar{\mathbf{Q}}_k$, $\bar{\mathbf{R}}_k$ and $\bar{\mathbf{B}}_k$ from the previous ones, $\mathbf{Q}_k$, $\mathbf{R}_k$ and $\mathbf{B}_k$.

Consider the updating $\bar{\mathbf{Q}}_k$ and $\bar{\mathbf{R}}_k$ from $\mathbf{Q}_k$ and $\mathbf{R}_k$ firstly. This technique is well described in [10]. Let $\mathbf{q}_1^T$ be the first row of $\mathbf{Q}_k$ and compute the Givens rotations $\mathbf{G}_1,\mathbf{G}_2,\cdots,\mathbf{G}_{n-2}$ such that

$$\mathbf{G}_1^T\mathbf{G}_2^T\cdots\mathbf{G}_{n-2}^T\mathbf{q}_1=\rho\mathbf{e}_1, \tag{16}$$

where $\rho=\pm1$. Note that

$$\mathbf{G}_1^T\mathbf{G}_2^T\cdots\mathbf{G}_{n-2}^T\mathbf{R}_k=\begin{bmatrix}\mathbf{v}^T\\ \bar{\mathbf{R}}_k\end{bmatrix} \tag{17}$$

is upper Hessenberg and that

$$\mathbf{Q}_k\mathbf{G}_{n-2}\cdots\mathbf{G}_1=\begin{bmatrix}\rho & 0\\ 0 & \bar{\mathbf{Q}}_k\end{bmatrix}, \tag{18}$$

where $\bar{\mathbf{Q}}_k\in\Re^{(n-2)\times(n-2)}$ is orthogonal. Thus, we can obtain the updated QR factorization.

Next, we consider the computation of $\bar{\mathbf{B}}_k$ from $\mathbf{B}_k$. Let

$$\left[\mathbf{A}_{k,2};\,\mathbf{b}_k\right]=\begin{bmatrix}\boldsymbol{\alpha}_{k+1}{}^T & \beta_{k+1}\\ \bar{\mathbf{A}}_{k,2} & \bar{\mathbf{b}}_k\end{bmatrix}, \tag{19}$$

where $\boldsymbol{\alpha}_{k+1}$ and β_{k+1} denote the row related to the (*k*+1)-th satellite.

Then,

$$\mathbf{G}_1^T\mathbf{G}_2^T\cdots\mathbf{G}_{n-2}^T\mathbf{B}_k=\mathbf{G}_1^T\mathbf{G}_2^T\cdots\mathbf{G}_{n-2}^T\mathbf{Q}_k^T\left[\mathbf{A}_{k,2};\,\mathbf{b}_k\right]$$

$$= \begin{bmatrix} \rho & 0 \\ 0 & \bar{\mathbf{Q}}_k^T \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{k+1}{}^T & \beta_{k+1} \\ \bar{\mathbf{A}}_{k,2} & \bar{\mathbf{b}}_k \end{bmatrix}$$
$$= \begin{bmatrix} \rho\boldsymbol{\alpha}_{k+1}{}^T & \rho\beta_{k+1} \\ \bar{\mathbf{Q}}_k^T \bar{\mathbf{A}}_{k,2} & \bar{\mathbf{Q}}_k^T \bar{\mathbf{b}}_k \end{bmatrix} \quad (20)$$
$$= \begin{bmatrix} \rho\boldsymbol{\alpha}_{k+1}{}^T ; \rho\beta_{k+1} \\ \bar{\mathbf{B}}_k \end{bmatrix}.$$

Thus, we can easily compute $\bar{\mathbf{B}}_k$ from $\mathbf{B}_k$.

As the second step, consider adding the previously deleted satellite. From (14) and (15), we have

$$\mathbf{H}_{k+1} \equiv diag\left(1, \bar{\mathbf{Q}}_k^T\right) \mathbf{P} \mathbf{A}_{k+1} = \begin{bmatrix} \boldsymbol{\alpha}_{k,1}^T; & \boldsymbol{\alpha}_{k,2}^T; \underline{\beta}_k \\ \bar{\mathbf{R}}_k \; ; & \bar{\mathbf{B}}_k \end{bmatrix}, \quad (21)$$

where

$$\mathbf{A}_{k+1} = \begin{bmatrix} \bar{\mathbf{A}}_{k,1}; & \bar{\mathbf{A}}_{k,2}; & \bar{\mathbf{b}}_k \\ \boldsymbol{\alpha}_{k,1}^T; & \boldsymbol{\alpha}_{k,2}^T; & \beta_k \end{bmatrix} \quad (22)$$

and the permutation matrix

$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ \mathbf{I}_{(n-2)\times(n-2)} & 0 \end{bmatrix}. \quad (23)$$

Note that $\mathbf{H}_{k+1}$ is upper Hessenberg in the first p columns. Thus, Givens rotations $\mathbf{J}_1, \mathbf{J}_2, \cdots, \mathbf{J}_p$ could be determined so

$$\mathbf{J}_p^T \mathbf{J}_{p-1}^T \cdots \mathbf{J}_1^T \begin{bmatrix} \boldsymbol{\alpha}_{k,1}^T \\ \bar{\mathbf{R}}_k \end{bmatrix} = \mathbf{R}_{k+1}, \quad \mathbf{R}_{k+1} \in \mathbf{R}^{(n-1)\times p} \quad (24)$$

is upper triangular. It follows that

$$\mathbf{J}_p \mathbf{J}_{p-1} \cdots \mathbf{J}_1 \mathbf{H}_{k+1} = \begin{bmatrix} \mathbf{R}_{k+1,11} & \mathbf{R}_{k+1,12} & \mathbf{y}_{k+1,1} \\ 0 & \mathbf{R}_{k+1,22} & \mathbf{y}_{k+1,2} \end{bmatrix}, (25)$$

where $\mathbf{R}_{k+1} = \begin{bmatrix} \mathbf{R}_{k+1,11} \\ 0 \end{bmatrix}$, $\mathbf{R}_{k+1,11} \in \mathbf{R}^{p\times p}$, $\mathbf{R}_{k+1,12} \in \mathbf{R}^{p\times(n-p)}$, $\mathbf{R}_{k+1,22} \in \mathbf{R}^{(n-p-1)\times(n-p)}$, $\mathbf{y}_{k+1,1} \in \mathbf{R}^p$ and $\mathbf{y}_{k+1,2} \in \mathbf{R}^{n-p-1}$. Thus, the original problem

$$\begin{bmatrix} \bar{\mathbf{A}}_{k,1}; & \bar{\mathbf{A}}_{k,2} \\ \boldsymbol{\alpha}_{k,1}^T; & \boldsymbol{\alpha}_{k,2}^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_{k+1,1} \\ \mathbf{x}_{k+1,2} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{b}}_k \\ \beta_k \end{bmatrix} \quad (26)$$

is converted to

$$\begin{bmatrix} \mathbf{R}_{k+1,11} & \mathbf{R}_{k+1,12} \\ 0 & \mathbf{R}_{k+1,22} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{k+1,1} \\ \mathbf{x}_{k+1,2} \end{bmatrix} = \begin{bmatrix} \mathbf{y}_{k+1,1} \\ \mathbf{y}_{k+1,2} \end{bmatrix} \quad (27)$$

without applying SVD to the original equation.

Then, we can solve the mixed LS-TLS problem separately. The TLS solution $\mathbf{x}_{k+1,2}$ of the equation

$$\mathbf{R}_{k+1,22} \mathbf{x}_{k+1,2} = \mathbf{y}_{k+1,2} \quad (28)$$

can be obtained by the SVD. Although the SVD is numerically stable, a great number of operations are required. To overcome this problem, we construct the following matrix

$$\mathbf{D}_{k+1} = \left[\mathbf{R}_{k+1,22}; \mathbf{y}_{k+1,2}\right]^T \left[\mathbf{R}_{k+1,22}; \mathbf{y}_{k+1,2}\right] \quad (29)$$

and compute the minimum eigenvector (associated with the minimum eigenvalue) using the FALM (Fast Algorithm to Locate Minimum eigenpair) [11,12] instead of computing the minimum singular vector of the matrix $R_{k+1,22}$ (associated with the minimum singular value). Using the TLS solution, the LS solution $\mathbf{x}_{k+1,1}$ of the equation

$$\mathbf{R}_{k+1,11} \mathbf{x}_{k+1,1} = \mathbf{y}_{k+1,1} - \mathbf{R}_{k+1,12} \mathbf{x}_{k+1,2} \quad (30)$$

can be obtained. Then the overall solution at the (k+1)-th stage becomes $\mathbf{x}_{k+1} = \left[\mathbf{x}_{k+1,1}^T; \mathbf{x}_{k+1,2}^T\right]^T$. For the next step, the $\mathbf{Q}_{k+1}$ is computed in a sequential form

$$\mathbf{Q}_{k+1} = \mathbf{P}^T diag\left(1, \bar{\mathbf{Q}}_k\right) \mathbf{J}_1^T \mathbf{J}_2^T \cdots \mathbf{J}_p^T. \quad (31)$$

Based on the above analysis, we summarize the algorithm below. The algorithm is expressed via MATLAB grammar, since it is simple and well known.

ALGORITHM

Given $\mathbf{Q}_k$, $\mathbf{R}_k$, $\mathbf{B}_k$, $\boldsymbol{\alpha}_k$, β_k

Step 1: Compute Givens rotations $\mathbf{G}_1, \mathbf{G}_2, \cdots, \mathbf{G}_{n-2}$ such that

$$\mathbf{G}_1^T \mathbf{G}_2^T \cdots \mathbf{G}_{n-2}^T \mathbf{q}_1 = \rho \mathbf{e}_1,$$

where $\mathbf{q}_1{}^T$ be the first row of $\mathbf{Q}_k$ and $\rho = \pm 1$.

Step 2: Compute $\bar{\mathbf{R}}_k$, $\bar{\mathbf{Q}}_k$ and $\bar{\mathbf{B}}_k$.

$$\bar{\mathbf{R}}_k = \mathbf{G}_1^T \cdots \mathbf{G}_{n-2}^T \mathbf{R}_k \left(2:(n-1),:\right)$$
$$\bar{\mathbf{Q}}_k = \mathbf{Q}_k \mathbf{G}_{n-2} \cdots \mathbf{G}_1 \left(2:(n-1), 2:(n-1)\right)$$
$$\bar{\mathbf{B}}_k = \mathbf{G}_1^T \cdots \mathbf{G}_{n-2}^T \mathbf{B}_k \left(2:(n-1),:\right)$$

Step 3: Compute Givens rotations $\mathbf{J}_1, \mathbf{J}_2, \cdots, \mathbf{J}_p$ such that

$$\mathbf{J}_p^T \mathbf{J}_{p-1}^T \cdots \mathbf{J}_1^T \begin{bmatrix} \boldsymbol{\alpha}_{k,1}^T \\ \bar{\mathbf{R}}_k \end{bmatrix} = \mathbf{R}_{k+1}, \qquad \mathbf{R}_{k+1} \in \mathbf{R}^{(n-1)\times p}$$

is upper triangular.

Step 4: Compute $\mathbf{R}_{k+1,12}$, $\mathbf{R}_{k+1,22}$, $\mathbf{y}_{k+1,1}$,

$\mathbf{y}_{k+1,2}$

$$\begin{bmatrix} \mathbf{R}_{k+1,12} & \mathbf{y}_{k+1,1} \\ \mathbf{R}_{k+1,22} & \mathbf{y}_{k+1,2} \end{bmatrix} = \mathbf{J}_p^T \mathbf{J}_{p-1}^T \cdots \mathbf{J}_1^T \begin{bmatrix} \boldsymbol{\alpha}_{k,2}^T ;\ \beta_k \\ \bar{\mathbf{B}}_k \end{bmatrix},$$

where $\mathbf{R}_{k+1,12} \in \mathbf{R}^{p\times(n-p)}$, $\mathbf{R}_{k+1,22} \in \mathbf{R}^{(n-p-1)\times(n-p)}$, $\mathbf{y}_{k+1,1} \in R^p$ and $\mathbf{y}_{k+1,2} \in R^{n-p-1}$.

Step 5: Construct the matrix $\mathbf{D}_{k+1}$ in (29) and compute the minimum eigenvector $\mathbf{v}$ of the matrix $\mathbf{D}_{k+1}$ using the FALM. Compute, then, the TLS solution $\mathbf{x}_{k+1,2}$

$$\mathbf{x}_{k+1,2} = -\frac{1}{v_{k-p+1}} \left[v_1, v_2, \cdots, v_{k-p} \right]^T$$

Step 6: Compute the least squares solution $\mathbf{x}_{k+1,1}$ of the equation (30). Then the overall solution is $\mathbf{x}_{k+1} = \left[\mathbf{x}_{k+1,1}^T ; \mathbf{x}_{k+1,2}^T \right]^T$.

Step 7: Compute $\mathbf{Q}_{k+1}$ as in (31). If every satellite is excluded one by one, stop. If not, proceed to *Step* 1.

Since to compute a Givens rotation matrix requires 4 flops and one square root and to multiply the Givens rotation matrix to a vector requires 4(m-1) flops [10], the algorithm requires about 44m+81 multiplications and 2m square roots for each subset test. The m is n-1, where n is the number of visible satellites. On the other hand, the SVD requires 50m + 256 for computation of singular values and right singular vectors and requires $10m^2 + 100m + \frac{14}{3}n^3$ flops for computation of singular values and right and left singular vectors.

4. SIMULATION RESULTS

In this section, a simulation result is discussed. The focus is on how well the proposed algorithm works under a satellite in a failed circumstance. The satellites data were generated using MATLAB toolbox [13]. Thermal noise, tropospheric error, multipath error, and ionospheric error were considered in generating the satellite data. A simulated pseudo-range error (1000m) was injected to a satellite at time t. Various magnitudes of the simulated error were used and they show similar results regardless of the magnitude. The value of the time t is not important because the proposed algorithm runs between adjacent epochs. In this simulation, we assume that there is no failure until time t-1 and a satellite (PRN #7 in this case) fails between t-1 and t. Then, the satellite date at time t is incorrect. We examine how the algorithm is working in this case. The following figures describe the results.

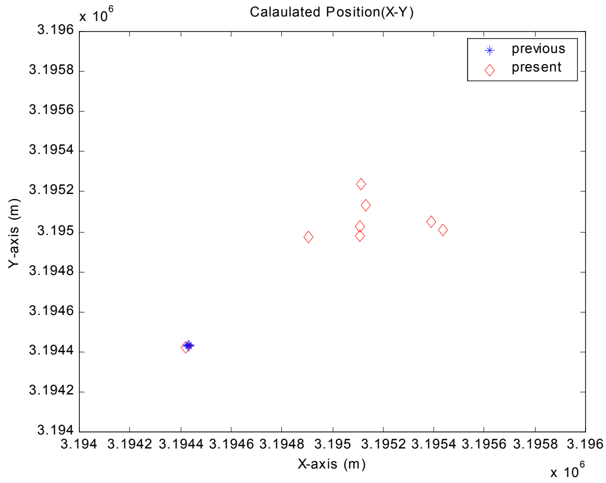

Fig. 1. Calculated position.

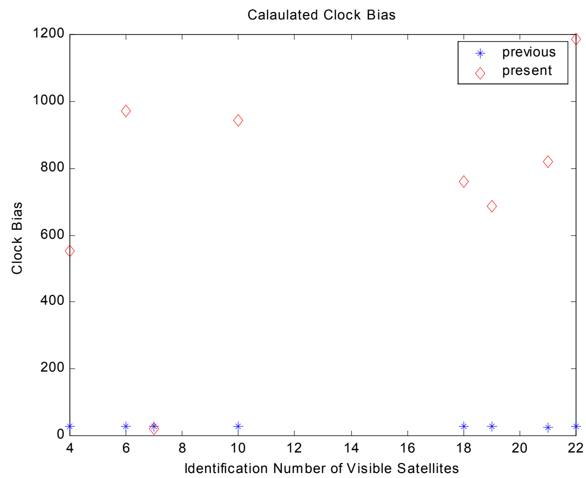

Fig. 2. Calculated clock bias.

Fig. 1 shows the calculated positions. The x-axis displays longitude and the y-axis denotes latitude. The star denotes previous positions at time t-1. Since the algorithm calculates a position with 7 satellites, 8 positions are calculated at each time. The 8 positions almost coincide with each other because there is no failure in 8 visible satellites.

The diamond denotes the present 8 positions. 7 present positions are apart from the previous position, which implies there is an errant satellite. Only one position is located near the previous positions. This means that the excluded satellite is failed.

Fig. 2 shows calculated clock bias. We assumed zero clock bias in this simulation. The 8 previous clock biases (star mark) are near zero. The diamonds stand for present clock biases. In the figure only one present clock bias (PRN #7) is near zero, which means the 7^{th} satellite is errant. With a proper measure and threshold even though it is not the focus of this paper, the proposed algorithm can provide superior performance for failure detection and identification based on TLS technique.

5. CONCLUSIONS

In this paper, a new TLS-based sequential algorithm to identify an errant satellite is proposed. A major contribution of this paper might be the fact that the algorithm is new and allows us to enjoy the advantages of TLS with less computational burden since it takes a sequential form. With a proper measure and threshold, which have been extensively studied until now, it can provide performance for failure detection and identification.

REFERENCES

[1] R. G. Brown, Receiver Autonomous Integrity Monitoring, In Bradford W. Parkinson and James J. Spilker, (Eds), *Global Position System: Theory and Applications,* vol. II, New York:AIAA, 1996.

[2] R. J. Kelly, "The linear model, RNP, and the near-optimum fault detection and exclusion algorithm", *Global Positioning System*, vol. V, ION, pp.227-259, 1998,.

[3] B. W. Parkinson and J. J. Spilker, (Eds), *Global Position System: Theory and Applications*, vol. I, New York:AIAA, 1996.

[4] D. Wells, et al, *Guide to GPS Positioning*, Canadian GPS Associates, 1987.

[5] Y. C. Lee, "Analysis of range and position comparison methods as a means to provide GPS integrity in the user receiver," *Proc. of the Annual Meeting of the Institute of Navigation (Seattle, WA), June 24-26*, pp. 1-4, 1986.

[6] B. W. Parkinson and P. Axelard, "Autonomous GPS integrity monitoring using the pseudorange residual," *Navigation (Washington)*, vol. 35, no. 2, pp. 255-274, 1988.

[7] B. S. Pervan, et al, "Parity space methods for autonomous fault detection and exclusion algorithms using carrier phase," *Proc. of PLANS'96*, 1996.

[8] J.-C. Juang, "On GPS position and integrity monitoring", *IEEE Trans. on Aerospace and Electronics Systems*, vol. 36, no. 1, pp. 327-336, January 2000.

[9] S. Van Huffel, and J. Vandewalle, *The Total Least Squares Problem Computational Aspects and Analysis*, SIAM, 1991.

[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins University Press, 1983.

[11] C. W. Jeon, H. J. Kim, and J. G. Lee, "A fast and accurate algorithm for computing desired eigenpairs of hermitian matrices," *IEICE Trans. on Information and Systems*, vol. E79-D, no. 3, pp. 182-188, March 1996.

[12] C. W. Jeon and J. G. Lee, "A new MRQI eigenpairs," *IEICE Trans. on Information and Systems*, vol. E82-D, no. 6, pp. 1011-1019, June 1999.

[13] GPSoft, *Satellite Navigation TOOLBOX 2.0 User's Guide*, GPSoft LLC, October 1999.

Chang-Wan Jeon received his B.S., M.S., and Ph.D. degrees in Control and Instrumentation Engineering from Seoul National University, Seoul, Korea, in 1986, 1988, 1996, respectively. In 1996, he was employed by the Samsung Electronic Company (SEC) undertaking development in the areas of vehicle navigation systems and integrated information and communication systems for the Incheon International Airport. In 1998, Dr. Jeon joined the Faculty of the Division of Information Technology Engineering, Soonchunhyang University, where he has been an Associate Professor since 2004. His research interests include system identification, Global Positioning System, Kalman filtering, and control system design.

Gyu-In Jee is a Professor in the Department of Electronics Engineering at Konkuk University, Seoul, Korea. He received his B.S. and M.S. degrees in Control and Instrumentation Engineering from Seoul National University and a Ph.D. in Systems Engineering from Case Western Reserve University. His research interests include GPS/INS integration, GPS receiver signal processing and Wireless Location. He is a Committee Member of the Korea GNSS Technical Council and Chairman of the TTA LBS Wireless Location Forum.

Gérard Lachapelle received his B.Sc. in Surveying from Laval University, a M.Sc. in Geodesy from Oxford, a L.Ph from the University of Helsinki, and a Dr.Techn from the Technical University of Graz. He is the CRC/iCORE Chair in Wireless Location in the Department of Geomatics Engineering at the University of Calgary where he has been a Professor since 1988. He has been involved in GNSS R&D since 1980 and has contributed to numerous aspects of ground and satellite-based navigation, including cellular telephone location. He holds degrees from Laval University, the University of Oxford, the University of Helsinki and the Technical University at Graz. Professor Lachapelle has been active in numerous associations and was Western Vice President of the (U.S.) Institute of Navigation in 2002-03. He has received scores of awards for his work, including the Institute of Navigation Johannes Kepler Award in 1997, and Fellowship of the Royal Society of Canada and the Institute of Navigation.