

Research Article

Robust Modeling of Low-Cost MEMS Sensor Errors in Mobile Devices Using Fast Orthogonal Search

M. Tamazin,¹ A. Noureldin,^{1,2} and M. J. Korenberg¹

¹ *Electrical and Computer Engineering Department, Queen's University, Kingston, ON, Canada K7L 3N6*

² *Electrical and Computer Engineering Department, Royal Military College of Canada, Kingston, ON, Canada K7K 7B4*

Correspondence should be addressed to M. Tamazin; tamazin@gmail.com

Received 4 February 2013; Accepted 23 May 2013

Academic Editor: Xiaoji Niu

Copyright © 2013 M. Tamazin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Accessibility to inertial navigation systems (INS) has been severely limited by cost in the past. The introduction of low-cost microelectromechanical system-based INS to be integrated with GPS in order to provide a reliable positioning solution has provided more wide spread use in mobile devices. The random errors of the MEMS inertial sensors may deteriorate the overall system accuracy in mobile devices. These errors are modeled stochastically and are included in the error model of the estimated techniques used such as Kalman filter or Particle filter. First-order Gauss-Markov model is usually used to describe the stochastic nature of these errors. However, if the autocorrelation sequences of these random components are examined, it can be determined that first-order Gauss-Markov model is not adequate to describe such stochastic behavior. A robust modeling technique based on fast orthogonal search is introduced to remove MEMS-based inertial sensor errors inside mobile devices that are used for several location-based services. The proposed method is applied to MEMS-based gyroscopes and accelerometers. Results show that the proposed method models low-cost MEMS sensors errors with no need for denoising techniques and using smaller model order and less computation, outperforming traditional methods by two orders of magnitude.

1. Introduction

Presently, GPS-enabled mobile devices offer various positioning capabilities to pedestrians, drivers, and cyclists. GPS provides absolute positioning information, but when signal reception is attenuated and becomes unreliable due to multipath, interference, and signal blockage, augmentation of GPS with inertial navigation systems (INS) or the like is needed. INS is inherently immune to the signal jamming, spoofing, and blockage vulnerabilities of GPS, but the accuracy of INS is significantly affected by the error characteristics of the inertial sensors it employs [1].

GPS/INS integrated navigation systems are extensively used [2], for example, in mobile devices that require low-cost microelectromechanical System (MEMS) inertial sensors (gyroscopes and accelerometers) due to their low cost, low power consumption, small size, and portability. The inadequate long-term performance of most commercially available MEMS-based INS limits their usefulness in providing reliable navigation solutions. MEMSs are challenging in any

consumer navigation system because of their large errors, extreme stochastic variance, and quickly changing error characteristics.

According to [3], the inertial sensor errors of a low-cost INS consist of two parts: a deterministic part and a random part. The deterministic part includes biases and scale factors, which are determined by calibration and then removed from the raw measurements. The random part is correlated over time and is basically due to the variations in the INS sensor bias terms. These errors are mathematically integrated during the INS mechanization process, which results in increasingly inaccurate position and attitude over time. Therefore, these errors must be modeled.

The fusion of INS and GPS data is a highly synergistic coupling as INS can provide reliable short-term positioning information during GPS outages, while GPS can correct for longer-term INS errors [1]. INS and GPS integration (i.e., data fusion) is typically achieved through an optimal estimation technique, such as the Kalman filter or Particle filter [4].

Despite having an INS/GPS integration algorithm to correct for INS errors, it is still advantageous to have an accurate INS solution before the data fusion process. This requires pre-processing (i.e., prefiltering or denoising) each of the inertial sensor (gyroscope and accelerometer) signals before they are used to compute position, velocity, and attitude. This paper offers a robust method based on fast orthogonal search (FOS) to model the stochastic errors of low-cost MEMS sensors for smart mobile phones.

Orthogonal search [5] is a technique developed for identifying difference equation and functional expansion models by orthogonalizing over the actual data record. It mainly utilizes Gram-Schmidt orthogonalization to create a series of orthogonal functions from a given set of arbitrary functions. This enables signal representation by a functional expansion of arbitrary functions and therefore provides a wider selection of candidate functions that can be used to represent the signal. FOS is a variant of orthogonal search [6] where one major difference is that FOS achieves orthogonal identification without creating orthogonal functions at any stage of the process. As a result FOS is many times faster and less memory storage intensive than the earlier technique, while equally as accurate and robust [5–7].

Many techniques have been used previously to denoise and stochastically model the inertial sensor errors [3, 8, 9]. For example, several levels of wavelet decomposition have been used to denoise the raw INS data and eliminate high-frequency disturbances [3, 8, 9]. Modeling inertial sensor errors using autoregressive (AR) models was performed in [3], where the Yule-Walker, the covariance, and Burg AR methods were used. The AR model parameters were estimated after reducing the INS sensor measurements noise using wavelet denoising techniques.

FOS has been applied before in several applications [5–7, 10–12]. In [13], FOS was used to augment a Kalman filter (KF) to enhance the accuracy of a low-cost 2D MEMS-based navigation system by modeling only the azimuth error. FOS is used in this paper to model the raw MEMS gyroscope and accelerometer measurement errors in the time domain. In this paper, the performance of FOS is compared to linear modeling techniques such as Yule-Walker, the covariance and Burg AR methods in terms of mean-square errors (MSEs) and computational time.

2. Problem Statement

It is generally accepted that the long-term errors are modeled as correlated random noise. Correlated random noise is typically characterized by an exponentially decaying autocorrelation function with a finite correlation time. When the autocorrelation function of some of the noise sequences of MEMS measurements is studied, it has been shown that a first-order Gauss-Markov (GM) process may not be adequate in all cases to model such noise behavior. The shape of the autocorrelation sequence is often different from that of a first-order GM process, which is represented by a decaying exponential as shown in Figure 1. The GM process is characterized by an autocorrelation function of the form $R_{xx}(\tau) = \sigma^2 e^{-\beta|\tau|}$, where σ^2 is variance of the process and the correlation time

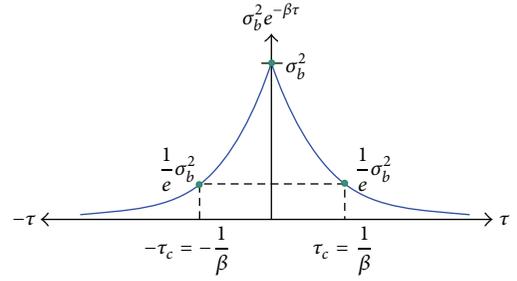


FIGURE 1: The autocorrelation sequence of a first-order Gauss-Markov (GM) process.

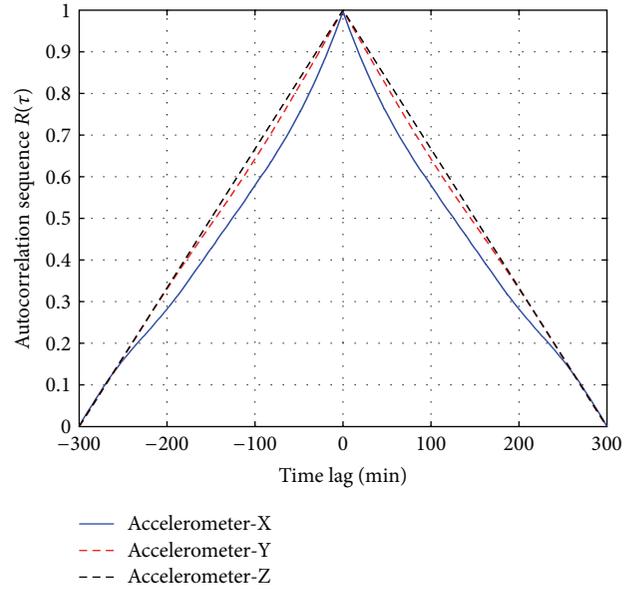


FIGURE 2: The computed autocorrelation sequence for an MEMS accelerometer data.

($1/e$ point) is given by $1/\beta$. The autocorrelation function approaches zero as $\tau \rightarrow \infty$, as depicted in Figure 1, indicating that the process gradually becomes less and less correlated as the time separation between samples increases [1].

Most of the computed autocorrelation sequences follow higher-order GM processes. An example of such computed autocorrelation sequences for one hour of static data of an MEMS accelerometer is shown in Figure 2. It clearly represents a higher-order GM process. These higher-order GM processes can be modeled using an autoregressive (AR) process of an appropriate order. In [3] it has been decided to model the randomness of the inertial sensor measurements using an AR process of order higher than one. With the present computational efficiency of microprocessor systems, efficient modeling of MEMS residual biases can be realized, and, thus, accurate prediction and estimation of such errors can be provided.

3. Modeling Methods of AR Processes

The autoregressive moving average (ARMA) modeling is based on the mathematical modeling of a time series of

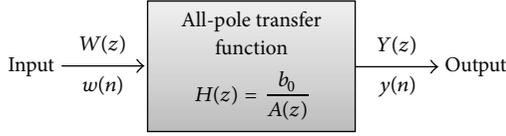


FIGURE 3: The input-output relationship of an autoregressive (AR) process.

measurements assuming that each value of such series is dependent on (a) a weighted sum of the “previous” values of the same series (AR part) and (b) a weighted sum of the “present and previous” values of a different time series (MA part) [14]. The ARMA process can be described using a pole-zero (AR-MA) transfer function system $H(z)$ as follows:

$$H(z) = \frac{Y(z)}{W(z)} = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^q b_k z^{-k}}{1 + \sum_{k=1}^p a_k z^{-k}}, \quad (1)$$

where $W(z)$ is the z -transform of the input $w(n)$, $Y(z)$ is the z -transform of the output $y(n)$, p is the order of the AR process, q is the order of the MA process, and a_1, a_2, \dots, a_p and b_1, b_2, \dots, b_q are the AR and MA process parameters (weights), respectively. The AR process is a special case of an ARMA process, where q in (1) will be zero and thus $H(z)$ will be only an all-pole transfer function of the form

$$H(z) = \frac{Y(z)}{W(z)} = \frac{B(z)}{A(z)} = \frac{b_0}{1 + \sum_{k=1}^p a_k z^{-k}}. \quad (2)$$

Therefore, the name “Autoregressive” comes from the fact that each signal sample is regressed on (or predicted from) the previous values of itself [3]. In the time domain, the previous AR transfer function relationship can be obtained after applying the inverse z -transform for (2).

The resultant equation is written as [14]

$$\begin{aligned} y(n) &= -\sum_{k=1}^p a_k y(n-k) + b_0 w(n) \\ y(n) &= -a_1 y(n-1) - a_2 y(n-2) \\ &\dots - a_p y(n-p) + b_0 w(n). \end{aligned} \quad (3)$$

The previous input-output relationship in both frequency and time domains is shown in Figure 3.

The problem in this case is to determine the values of the AR model parameters (predictor coefficients) a_k that optimally represent the random part of the inertial sensor biases. This is performed by minimizing the error $e(n)$ between the original signal $y(n)$ represented by the “AR process” of (3) and the estimated signal $\hat{y}(n)$, which is estimated by an “AR model” of the form [8]

$$\hat{y}(n) = -\sum_{k=1}^p a_k y(n-k). \quad (4)$$

The cost function for this minimization problem is the energy E of $e(n)$, which is given as

$$\begin{aligned} E &= \sum_{n=1}^N e^2(n) = \sum_{n=1}^N [y(n) - \hat{y}(n)]^2 \\ &= \sum_{n=1}^N \left[-\sum_{k=1}^p a_k y(n-k) + b_0 w(n) + \sum_{k=1}^p a_k y(n-k) \right]^2 \quad (5) \\ &= \sum_{n=1}^N b_0^2 w^2(n) = \min, \end{aligned}$$

where N is the total number of data samples. In this case, $w(n)$ is a sequence of stationary uncorrelated sequences (white noise) with zero mean and unity variance.

Therefore, the resultant energy from (5) $[\sum_{n=1}^N b_0^2 w^2(n)]$ will be b_0^2 . Therefore, b_0^2 represents the estimated variance σ_w^2 of the white noise input to the AR model or, more generally, the prediction mean-square error σ_e^2 . This is due to the fact that the AR model order p is completely negligible with respect to the MEMS data sample size N .

Several methods have been reported to estimate the a_k parameter values by fitting an AR model to the input data. Three AR methods are considered in this paper, namely, the Yule-Walker method, the covariance method, and Burg’s method. In principle, all of these estimation techniques should lead to approximately the same parameter values if fairly large data samples are used [3].

3.1. The Yule-Walker Method. The Yule-Walker method, which is also known as the autocorrelation method, determines first the autocorrelation sequence $R(\tau)$ of the input signal (inertial sensor residual bias in our case). Then, the AR model parameters are optimally computed by solving a set of linear normal equations. These normal equations are obtained using the formula [15]

$$\frac{\partial E}{\partial a_k} = 0, \quad (6)$$

which leads to the following set of normal equations:

$$Ra = -r \iff a = -R^{-1}r, \quad (7)$$

where

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \quad (8a)$$

$$r = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix}, \quad (8b)$$

$$R = \begin{bmatrix} R(0) & R(1) & \dots & R(p-1) \\ R(1) & R(0) & \dots & R(p-2) \\ \vdots & \vdots & \dots & \vdots \\ R(p-1) & R(p-2) & \dots & R(0) \end{bmatrix}. \quad (8c)$$

If the mean-square error σ_e^2 is also required, it can be determined by

$$\begin{aligned} & \begin{bmatrix} R(0) & R(1) & \dots & R(p-1) \\ R(1) & R(0) & \dots & R(p-2) \\ \vdots & \vdots & \dots & \vdots \\ R(p-1) & R(p-2) & \dots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} \\ & = \begin{bmatrix} \sigma_e^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \end{aligned} \quad (9a)$$

Equations (7) and (9a) are known as the Yule-Walker equations [7–9, 13]. Instead of solving (9a) directly (i.e., by first computing R^{-1}), it can efficiently be solved using the Levinson-Durbin (LD) algorithm which proceeds recursively to compute a_1, a_2, \dots, a_p , and σ_e^2 . The LD algorithm is an iterative technique that computes the next prediction coefficient (AR parameter) from the previous one. This LD recursive procedure can be summarized in the following [9]:

$$E_0 = R(0) \quad (9b)$$

$$\gamma_k = -\frac{R(k) + \sum_{i=1}^{k-1} a_{i,k-1} R(k-i)}{E_{k-1}}, \quad 1 \leq k \leq p \quad (9c)$$

$$a_{k,k} = \gamma_k \quad (9d)$$

$$a_{i,k} = a_{i,k-1} + \gamma_k a_{k-i,k-1}, \quad 1 \leq i \leq k-1 \quad (9e)$$

$$E_k = (1 - \gamma_k^2) E_{k-1}. \quad (9f)$$

Equations (9b)–(9f) are solved recursively for $k = 1, 2, \dots, p$ and the final solution for the AR parameters is provided by

$$a_i = a_{i,p}, \quad 1 \leq i \leq p. \quad (9g)$$

Therefore, the values of the AR prediction coefficients in the Yule-Walker method are provided directly based on minimizing the forward prediction error $e_f(n)$ in the least-squares sense. The intermediate quantities γ_k represented by (9c) are known as the reflection coefficients. In (9f), both energies E_k and E_{k-1} are positive, and, thus, the magnitude of γ_k should be less than one to guarantee the stability of the all-pole filter.

However, the Yule-Walker method performs adequately only for long data records [15]. The inadequate performance in case of short data records is usually due to the data windowing applied by the Yule-Walker algorithm. Moreover, the Yule-Walker method may introduce a large bias in the AR-estimated coefficients since it does not guarantee a stable solution of the model [16].

3.2. The Covariance Method. The covariance method is similar to the Yule-Walker method in that it minimizes the forward prediction error in the least-squares sense, but it does

not consider any windowing of the data. Instead, the windowing is performed with respect to the prediction error to be minimized. Therefore, the AR model obtained by this method is typically more accurate than the one obtained from the Yule-Walker method [17].

Furthermore, it uses the covariance $C(\tau_i, \tau_j)$ instead of $R(\tau)$. In this case, the Toeplitz structure of the normal equations used in the autocorrelation method is lost, and hence the LD algorithm cannot be used for the computations. To achieve an efficient C^{-1} in this case, Cholesky factorization is usually utilized [15].

The method provides more accurate estimates than the Yule-Walker method especially for short data records. However, the covariance method may lead to unstable AR models since the LD algorithm is not used for solving the covariance normal equations [18].

3.3. Burg's Method. Burg's method was introduced in 1967 to overcome most of the drawbacks of the other AR modeling techniques by providing both stable models and high resolution (i.e., more accurate estimates) for short data records [19]. Burg's method tries to make the maximum use of the data by defining both a forward and a backward prediction error terms, $e_f(n)$ and $e_b(n)$. The energy to be minimized in this case (E_{Burg}) is the sum of both the forward and backward prediction error energies; that is,

$$E_{\text{Burg}} = \sum_{n=1}^N [e_f^2(n) + e_b^2(n)] = \min, \quad (10)$$

where e_f and e_b are defined as

$$e_f = y(n) + a_1 y(n-1) + a_2 y(n-2) + \dots + a_p y(n-p) \quad (11a)$$

$$e_b = y(n-p) + a_1 y(n-p+1) + a_2 y(n-p+2) + \dots + a_p y(n). \quad (11b)$$

The forward and backward prediction error criteria are the same, and, hence, they have the same optimal solution for the model coefficients [20]. Considering the energies in (9f) to be E_{Burg} , the forward and backward prediction errors can, therefore, be expressed recursively as

$$e_{f^k}(n) = e_{f^{k-1}}(n) + \gamma_k e_{b^{k-1}}(n-1) \quad (12a)$$

$$e_{b^k}(n) = e_{b^{k-1}}(n-1) + \gamma_k e_{f^{k-1}}(n). \quad (12b)$$

These recursion formulas form the basis of what is called Lattice (or Ladder) realization of a prediction error filtering (see Figure 4).

As has been shown for the Yule-Walker method, the accuracy of the estimated parameters a_1, a_2, \dots, a_p , and σ_e^2 depends mainly on accurate estimates of the autocorrelation sequence $R(\tau)$. However, this can be rarely achieved due to the prewindowing of data [17] or the existence of large

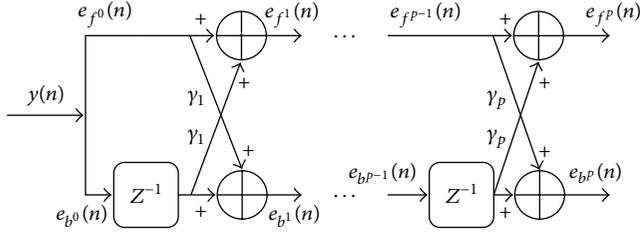


FIGURE 4: The forward-backward prediction error lattice filter general structure.

measurement noise [21]. To avoid the difficulties of the computation of the autocorrelation sequences, Burg in his method estimates first the reflection coefficients γ_k using another formula instead of (9c). This formula is derived by substituting (12a) and (12b) into (13) and setting the derivative of E_{Burg} with respect to γ_k (instead of a_k in the Yule-Walker and covariance methods) to zero.

This leads to the form

$$\gamma_k = \frac{-2 \sum_{n=1}^N [e_{f^{k-1}}(n) e_{b^{k-1}}(n-1)]}{\sum_{n=1}^N e_{f^{k-1}}^2(n) + \sum_{n=1}^N e_{b^{k-1}}^2(n-1)} \quad (13)$$

which shows clearly that the magnitude of γ_k is forced (guaranteed) to be less than one, and thus the obtained model is guaranteed to be stable. Equations (12a), (12b), and (13) form the recursive structure of Burg's lattice filter, which is shown in Figure 4 with the initial conditions of $e_f^0(n) = e_b^0(n) = y(n)$. Finally, the prediction coefficients a_k are obtained by constraining them to satisfy (9e) in the LD algorithm.

Therefore, the utilization of (9e) and (13) together will always ensure the stability of Burg's method solution [22]. Moreover, the utilization of both forward and backward prediction errors minimization usually yields better estimation results than using the forward prediction approach used in the previous two methods. Finally, it has been reported by [23] that Burg's method generally provides better residual estimates than the Yule-Walker method [19].

4. Fast Orthogonal Search (FOS) Method

FOS [5–7, 10–12] is a general purpose modeling technique, which can be applied to spectral estimation and time-frequency analysis. The algorithm uses an arbitrary set of nonorthogonal candidate functions $p_m(n)$ and finds a functional expansion of an input $y(n)$ in order to minimize the mean square error (MSE) between the input and the functional expansion.

The functional expansion of the input $y(n)$ in terms of the arbitrary candidate functions $p_m(n)$ is given by

$$y(n) = \sum_{m=0}^M a_m P_m(n) + e(n), \quad (14)$$

where a_m is the set of weights of the functional expansion, $P_0(n) = 1$, and the $P_m(n)$ are the model terms selected from the set of candidate functions, and $e(n)$ is the modeling error.

These model terms can involve the system input x and output y and cross-products and powers thereof:

$$P_m(n) = y(n-l_1) \cdots y(n-l_i) x(n-k_1) \cdots x(n-k_j), \\ m \geq 1, \quad i \geq 0, \quad j \geq 0, \quad \forall i: 1 \leq l_i \leq L, \quad \forall j: 0 \leq k_j \leq K. \quad (15)$$

By choosing non-orthogonal candidate functions, there is no unique solution for (14). However, FOS may model the input with fewer model terms than an orthogonal functional expansion [11]. For the FFT to model a frequency that does not have an integral number of periods in the record length, energy is spread into all the other frequencies, which is a phenomenon known as spectral leakage [24]. By using candidate functions that are non-orthogonal, FOS may be able to model this frequency between two FFT bins with a single term resulting in many fewer weighting terms in the model [5, 25].

FOS begins by creating a functional expansion using orthogonal basis functions such that

$$y(n) = \sum_{m=0}^M g_m w_m(n) + e(n), \quad (16)$$

where $w_m(n)$ is a set of orthogonal functions derived from the candidate functions $p_m(n)$, g_m is the weight, and $e(n)$ is an error term. The orthogonal functions $w_m(n)$ are derived from the candidate functions $p_m(n)$ using the Gram-Schmidt (GS) orthogonalization algorithm. The orthogonal functions $w_m(n)$ are implicitly defined by the Gram-Schmidt coefficients α_{mr} and do not need to be computed point-by-point.

The Gram-Schmidt coefficients α_{mr} and the orthogonal weights g_m can be found recursively using the equations [11]

$$w_0 = p_0(n) \quad (17)$$

$$D(m, 0) = \overline{p_m(n) p_0(n)} \quad (18)$$

$$D(m, r) = \overline{p_m(n) p_r(n)} - \sum_{i=0}^{r-1} \alpha_{ri} D(m, i) \quad (19)$$

$$\alpha_{mr} = \frac{\overline{p_m(n) w_r(n)}}{\overline{w_r^2(n)}} = \frac{D(m, r)}{D(r, r)} \quad (20)$$

$$C(0) = \overline{y(n) p_0(n)} \quad (21)$$

$$C(m) = \overline{y(n) p_m(n)} - \sum_{r=0}^{m-1} \alpha_{mr} C(r) \quad (22)$$

$$g_m = \frac{C(m)}{D(m, m)}. \quad (23)$$

In its last stage, FOS calculates the weights of the original functional expansion a_m (6), from the weights of the orthogonal series expansion g_m and Gram-Schmidt coefficients α_{mr} . The value of a_m can be found recursively using

$$a_m = \sum_{i=m}^M g_i v_i, \quad (24)$$

where $v_m = 1$ and

$$v_i = -\sum_{r=m}^{i-1} \alpha_{ir} v_r, \quad i = m+1, m+2, \dots, M. \quad (25)$$

FOS requires the calculation of the correlation between the candidate functions and the calculation of the correlation between the input and the candidate functions. The correlation between the input and the candidate function $\overline{y(n)p_m(n)}$ is typically calculated point-by-point once at the start of the algorithm and then stored for later quick retrieval.

The MSE of the orthogonal function expansion has been shown to be [5, 6, 11]

$$\overline{\varepsilon^2(n)} = \overline{y^2(n)} - \sum_{m=0}^M \overline{g_m^2 w_m^2(n)}. \quad (26)$$

It then follows that the MSE reduction given by the m th candidate function is given by

$$Q_m = g_m^2 \overline{w_m^2(n)} = g_m^2 D(m, m), \quad (27)$$

The candidate with the greatest value for Q is selected as the model term, but optionally its addition to the model may be subject to its Q value exceeding a threshold level [5, 6, 11]. The residual MSE after the addition of each term can be computed by

$$\text{MSE}_m = \text{MSE}_{m-1} - Q_m. \quad (28)$$

The search algorithm may be stopped when an acceptably small residual MSE has been achieved (i.e., a ratio of the MSE over the mean-squared value of the input [12] or an acceptably small percentage of the variance of the time series being modeled [5]). The search may also stop when a certain number of terms have been fitted. Another stopping criterion is when none of the remaining candidates can yield a sufficient MSE reduction value (this criterion would be representative of not having any candidates that would yield an MSE reduction value greater than the addition of a white Gaussian noise series).

5. Experimental Results

The data were collected by a low-cost MEMS-based inertial measurement unit (IMU CC-300, Crossbow). These measurements were collected during a one-hour experiment to obtain stochastic error models of both gyroscopes and accelerometers. To illustrate the performance, two sensors were selected as an example (accelerometer-Y, Gyro-Y) while the other inertial sensors gave similar results. Figure 5 shows one hour of sampled accelerometer-Y and Gyro-Y acquired at 200 Hz.

FOS is applied directly on the raw inertial sensor 200 Hz data without any preprocessing or denoising. Traditional methods like Yule Walker, Covariance, and Burg perform poorly on the raw data, so we first applied wavelet denoising of up to 4 levels of decomposition that resulted in band limiting the spectrum of the raw inertial sensor data to 12.5 Hz.

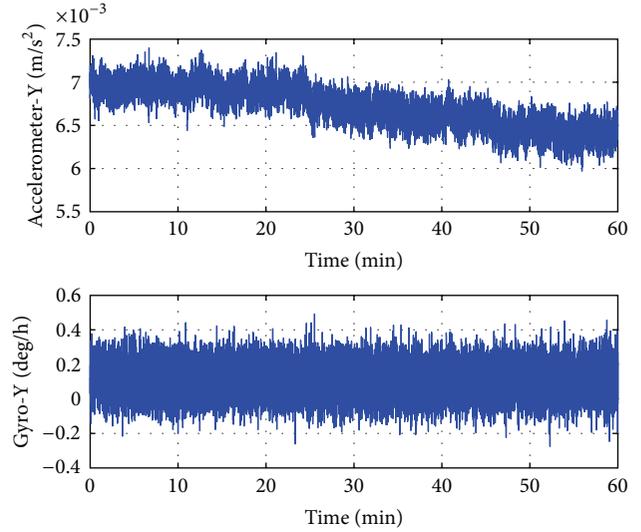


FIGURE 5: Accelerometer-Y and Gyro-Y specific force measurements.

Therefore, unlike FOS, the other 3 methods operate on the denoised version of the same data. After denoising, AR model parameters were then estimated as well as the corresponding prediction MSE for all sensors using Yule-Walker, Covariance, and Burg methods.

For FOS, the raw INS data were divided into three datasets for model training, evaluation, and prediction stages. The first 3 minutes of the INS raw data were utilized for model training, which uses the FOS algorithm to identify several possibly nonlinear AR equations. Different models can be obtained by changing the maximum delay L in the output and the degree of output cross-products (CP). The next 3 minutes of the data were used for the evaluation stage. Here, models are compared and the best one, fitting the real output with minimum MSE, is chosen. As an example, the FOS model (CP = 1) of the accelerometer-Y is shown as follows:

$$\begin{aligned} Y[n] = & 6.03 \times 10^{-6} + 2.25y[n-1] - 1.2y[n-2] \\ & + 1.02y[n-4] - 0.67y[n-3] - 0.26y[n-5] \\ & + 0.62y[n-7] - 0.51y[n-6] - 0.32y[n-8] \\ & + 0.08y[n-9]. \end{aligned} \quad (29)$$

In the prediction stage, the output and MSE of the chosen model are computed over the remaining (novel) raw INS data. Figure 6 shows prediction MSE of Accelerometer-Y samples by Yule-Walker, Covariance, Burg, and FOS methods. For FOS, an AR model of order 3 or 4 suffices, and MSE decreases when the degree of the cross-product terms is raised to 2 (nonlinear model). An AR model of order 7 or 8 is required for Burg or Covariance method and order 9 or 10 for Yule-Walker. Large AR model order complicates the estimation method (like KF) used for the INS/GPS integration.

Table 1 shows a summary of the performance of three conventional stochastic modeling methods (Yule-Walker,

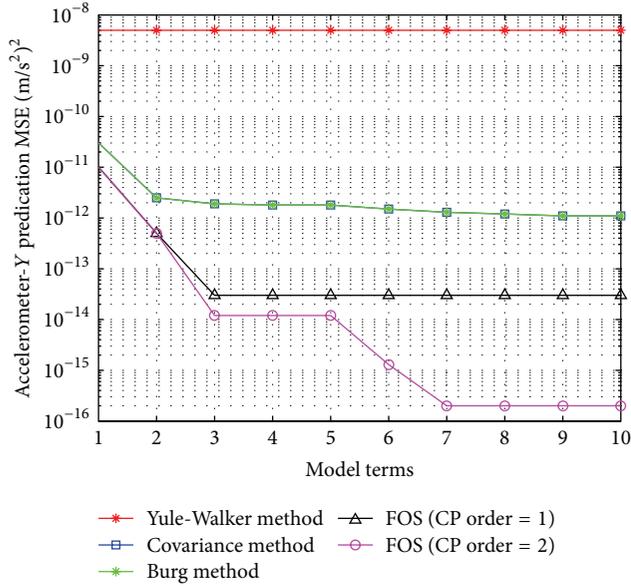


FIGURE 6: Accelerometer-Y prediction MSE using Yule-Walker, covariance, Burg, and FOS methods. For CP degree = 1, only linear candidate terms up to a maximum output lag $L = 10$ were allowed. For CP degree = 2, both linear and $y(n-l_1)y(n-l_2)$ candidates were allowed, up to maximum output lag $L = 10$.

TABLE 1: Performance summary of AR models obtained by Yule-Walker, Burg, and FOS over one hour of accelerometer-Y measurements: first-order FOS: only linear model terms; second-order FOS: linear and cross-product model terms.

Modelling technique	Model MSE (m/s^2) ²	Corresponding position error (m)	Computational time (s)
Model order (maximum output lag L) 1			
Yule-Walker	5×10^{-9}	458	0.25
Covariance/Burg	3×10^{-11}	35	0.23
CP degree = 1 FOS	1×10^{-11}	20	0.13
CP degree = 2 FOS	1×10^{-11}	20	0.36
Model order (maximum output lag L) 10			
Yule-Walker	5×10^{-9}	458	0.69
Covariance/Burg	2×10^{-12}	9	0.45
CP degree = 1 FOS	3×10^{-14}	1	0.30
CP degree = 2 FOS	2×10^{-16}	0.09	0.60

Covariance, and Burg) and the proposed FOS-based method with cross-product order set to 1 (i.e., linear model) and 2 (i.e., nonlinear model) for different model orders 1 and 10 over one hour of MEMS Accelerometer-Y measurements. The FOS model is capable of denoising the Accelerometer-Y measurements without appreciable degradation of the original signal. FOS achieves better performance in terms of lower MSE and less computational time than the traditional methods with no need for any denoising techniques. Increasing the cross-product degree to 2 for FOS improves model accuracy and

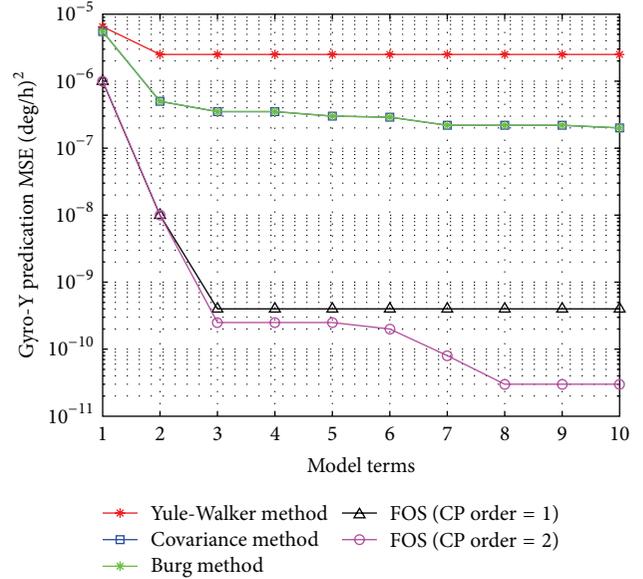


FIGURE 7: Gyro-Y prediction MSE using Yule-Walker, covariance, Burg, and FOS methods. For CP order = 1, only linear candidate terms up to a maximum output lag $L = 10$ were allowed. For CP order = 2, both linear and $y(n-l_1)y(n-l_2)$ candidates were allowed, up to a maximum output lag $L = 10$.

TABLE 2: Performance summary of both AR Yule-Walker and Burg models and FOS model over one hour of Gyro-Y measurements: first-order FOS: only linear model terms; second-order FOS: linear and cross-product model terms.

Modeling technique	Model MSE (deg/h) ²	Corresponding position error (m)	Computational time (s)
model order (maximum output lag L) 1			
Yule-Walker	7×10^{-6}	978	0.40
Covariance/Burg	5×10^{-6}	826	0.23
CP degree = 1 FOS	1×10^{-6}	369	0.09
CP degree = 2 FOS	1×10^{-6}	369	0.35
Model order (maximum output lag L) 10			
Yule-Walker	2×10^{-6}	523	0.44
Covariance/Burg	2×10^{-7}	165	0.28
CP degree = 1 FOS	4×10^{-10}	7	0.18
CP degree = 2 FOS	3×10^{-11}	2	0.6

lessens position error by an order of magnitude (for $L = 10$) but increases computation time.

A similar procedure was performed for the Gyro-Y sensor measurements. Figure 7 shows the prediction MSE of Gyro-Y samples by using Yule-Walker, the Covariance, Burg, and FOS methods. It is clear that FOS method achieves minimum MSE error with less model order than the other AR methods. Table 2 shows a summary of the performance of three conventional stochastic modeling methods (Yule-Walker, Covariance, and Burg) and the proposed FOS-based method with cross-product order set to 1 (i.e., linear model)

and 2 (i.e., nonlinear model) for different model orders 1 and 10 over one hour of Gyro-Y measurements.

Similar to the accelerometer case, the stochastic model obtained for Gyro-Y using FOS surpasses the models obtained by the other methods in the MSE, the corresponding position error that would result from the residual errors and the computation time.

6. Conclusions

Inertial sensor errors are the most significant contributors to INS errors. Thus, techniques to model these sensor errors are of interest to researchers. The current state of the art in modeling inertial sensor signals includes low-pass filtering and using wavelet denoising techniques, which have had limited success in removing long-term inertial sensor errors.

This paper suggested using FOS to model the MEMS sensor errors in time domain. The FOS MSE and computational time are compared with those from Yule-Walker, the covariance, and Burg methods. FOS was applied directly to the one-hour raw inertial sensor 200 Hz data without any pre-processing or denoising. The other traditional 3 methods operated on the denoised version of the same data after we applied the wavelet denoising of up to 4 levels of decomposition.

For either gyroscope or accelerometer case, the FOS model surpasses those obtained by traditional methods. The results demonstrate the advantages of the proposed FOS-based method including the absence of a need for pre-processing or denoising, lower computation time and MSE, and achieving better performance with smaller model order. Increasing the cross-product degree for FOS improves model accuracy and lessens position error but increases computation time.

References

- [1] A. Noureldin, T. Karamat, and J. Georgy, *Fundamentals of Inertial Navigation, Satellite-Based Positioning and Their Integration*, Springer, New York, NY, USA, 2012.
- [2] M. Grewal, L. Weil, and A. Andrews, *Global Positioning Systems, Inertial Navigation and Integration*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2007.
- [3] S. Nassar, K. Schwarz, N. El-Sheimy, and A. Noureldin, "Modeling inertial sensor errors using autoregressive (AR) models," *Navigation, Journal of the Institute of Navigation*, vol. 51, no. 4, pp. 259–268, 2004.
- [4] J. Georgy and A. Noureldin, "Vehicle navigator using a mixture particle filter for inertial sensors/odometer/map data/GPS integration," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 544–522, 2012.
- [5] M. J. Korenberg, "A robust orthogonal algorithm for system identification and time-series analysis," *Biological Cybernetics*, vol. 60, no. 4, pp. 267–276, 1989.
- [6] M. J. Korenberg, "Fast orthogonal identification of nonlinear difference equation models," in *Proceedings of the 30th Midwest Symposium on Circuits and Systems*, vol. 1, August 1987.
- [7] J. Armstrong, A. Noureldin, and D. McGaughey, "Application of fast orthogonal search techniques to accuracy enhancement of inertial sensors for land vehicle navigation," in *Proceedings of the Institute of Navigation, National Technical Meeting (NTM '06)*, pp. 604–614, Monterey, Calif, USA, January 2006.
- [8] A. Noureldin, A. Osman, and N. El-Sheimy, "A neuro-wavelet method for multi-sensor system integration for vehicular navigation," *Measurement Science and Technology*, vol. 15, no. 2, pp. 404–412, 2004.
- [9] S. Nassar, A. Noureldin, and N. El-Sheimy, "Improving positioning accuracy during kinematic DGPS outage periods using SINS/DGPS integration and SINS data de-noising," *Survey Review*, vol. 37, no. 292, pp. 426–438, 2004.
- [10] K. M. Adeney and M. J. Korenberg, "Fast orthogonal search for array processing and spectrum estimation," *IEE Proceedings: Vision, Image and Signal Processing*, vol. 141, no. 1, pp. 13–18, 1994.
- [11] M. J. Korenberg, "Fast orthogonal algorithms for nonlinear system identification and time-series analysis," in *Advanced Methods of Physiological System Modeling*, V. Z. Marmarelis, Ed., vol. 2, pp. 165–177, Plenum Press, New York, NY, USA, 1989.
- [12] D. R. McGaughey, M. J. Korenberg, K. M. Adeney, S. D. Collins, and G. J. M. Aitken, "Using the fast orthogonal search with first term reselection to find subharmonic terms in spectral analysis," *Annals of Biomedical Engineering*, vol. 31, no. 6, pp. 741–751, 2003.
- [13] Z. Shen, J. Georgy, M. J. Korenberg, and A. Noureldin, "FOS-based modelling of reduced inertial sensor system errors for 2D vehicular navigation," *Electronics Letters*, vol. 46, no. 4, pp. 298–299, 2010.
- [14] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Upper Saddle River, NJ, USA, 2002.
- [15] L. Jackson, *Digital Filters and Signal Processing*, Kluwer Academic, Norwell, Mass, USA, 1996.
- [16] R. Kless and P. Broersen, *How to Handle Colored Noise in Large Least-Squares Problems*, Delft University of Technology, Delft, The Netherlands, 2002.
- [17] M. Hayes, *Statistical Digital Signal Processing and Modeling*, John Wiley & Sons, New York, NY, USA, 1996.
- [18] M. J. L. de Hoon, T. H. J. van der Hagen, H. Schoonewelle, and H. van Dam, "Why Yule-Walker should not be used for autoregressive modelling," *Annals of Nuclear Energy*, vol. 23, no. 15, pp. 1219–1228, 1996.
- [19] J. Burg, *Maximum entropy spectral analysis [Ph.D. thesis]*, Department of Geophysics, Stanford University, Stanford Calif, USA, 1975.
- [20] S. Orfanidis, *Optimum Signal Processing: An Introduction*, Macmillan, New York, NY, USA, 1988.
- [21] J. M. Pimbley, "Recursive autoregressive spectral estimation by minimization of the free energy," *IEEE Transactions on Signal Processing*, vol. 40, no. 6, pp. 1518–1527, 1992.
- [22] S. Marple, *Digital Spectral Analysis with Applications*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1987.
- [23] I. A. Rezek and S. J. Roberts, "Parametric model order estimation: a brief review," in *Proceedings of the IEE Colloquium on the Use of Model Based Digital Signal Processing Techniques in the Analysis of Biomedical Signals*, London, UK, April 1997.
- [24] E. Ifeachor and B. Jervis, *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 4th edition, 2001.
- [25] K. H. Chon, "Accurate identification of periodic oscillations buried in white or colored noise using fast orthogonal search," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 6, pp. 622–629, 2001.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

