Particle swarm optimization for GPS navigation
Kalman filter adaptation

Dah-Jing Jwo
Department of Communications, Navigation and Control Engineering, National Taiwan Ocean University, Keelung, Taiwan, and
Shun-Chieh Chang
Prolific Technology, Hsinchu City, Taiwan

Abstract
Purpose – The purpose of this paper is to conduct the particle swarm optimization (PSO)-assisted adaptive Kalman filter (AKF) for global positioning systems (GPS) navigation processing. Performance evaluation for the PSO-assisted Kalman filter (KF) as compared to the conventional KF is provided.
Design/methodology/approach – The position-velocity also known as constant velocity process model can be applied to the GPS KF adequately when navigating a vehicle with constant speed. However, when an abrupt acceleration motion occurs, the filtering solution becomes very poor or even diverges. To avoid the limitation of the KF, the PSO can be incorporated into the filtering mechanism as dynamic model corrector. The PSO is utilized as the noise-adaptive mechanism to tune the covariance matrix of process noise and overcome the deficiency of KF. In other words, PSO-assisted KF approach is employed for tuning the covariance of the GPS KF so as to reduce the estimation error during substantial maneuvering.
Findings – The paper provides an alternative approach for designing an AKF and provides an example in the application to GPS.
Practical implications – The proposed scheme enhances the improvement in estimation accuracy. Application of the PSO to the GPS navigation filter design is discussed. The method takes advantage of both the adaptation capability and the robustness of numerical stability.
Originality/value – The PSO are employed for assisting the AKF. The use of optimization such as PSO for AKF has seldom been seen in the open literature.

Keywords Data communication systems, Velocity measurement, Particle physics, Navigation

Paper type Research paper

1. Introduction

The Kalman filter (KF) (Brown and Hwang, 1997; Gelb, 1974; Grewal and Andrews, 2001; Lewis, 1986; Jwo and Cho, 2007) is the most well known sequential data assimilation scheme for solving the Wiener problem in a generally easier way. It has been applied in the areas as diverse as aerospace, marine navigation, radar target tracking, control systems, manufacturing, and many others. A navigation filter is commonly designed by use of a KF to estimate the vehicle state variables and suppress the navigation measurement noise. The KF not only works well in practice, but also it is theoretically attractive because it has been shown that it is the filter that minimizes the variance of the estimation mean square error.

In practice, the KF attempts to minimize the variance of the estimation errors and provides optimal result if perfect a priori knowledge of both the process noise and measurement noise are Gaussian white processes and the noise statistics for both disturbances are completely known. To obtain good estimation solutions using the extended Kalman filter (EKF) approach, the designers are required to have good knowledge on both dynamic process (plant dynamics, using an estimated internal model of the dynamics of the system) and measurement models, in addition to the assumption that both the process and measurement are corrupted by zero-mean white noises.

In KF designs, the case that theoretical behavior of a filter and its actual behavior do not agree may lead to divergence problems. For example, if the KF is provided with information that the process behaves a certain way, whereas, in fact, it behaves a different way, the filter will continually intend to fit an incorrect process signal. Furthermore, when the measurement situation does not provide sufficient information to estimate all the state variables of the system, in other words, the estimation error covariance matrix becomes unrealistically small and the filter disregards the measurement. In various circumstances where there are uncertainties in the system model and noise description, and the assumptions on the statistics of disturbances are violated since in a number of practical situations, the availability of a precisely known model is unrealistic due to the fact that in the modeling step, some phenomena are disregarded and a way to take them into account is to consider a nominal model affected by uncertainty.

These assumptions mentioned above limit the application of minimum variance estimators in different situations where there are uncertainties in the system model and noise description, and the assumptions on the statistics of disturbances are violated. The implementation of KF requires the a priori knowledge of both the process and
measurement models. In a number of practical situations, the availability of a precisely known model is unrealistic because, in the modelling step, some phenomena are disregarded and a way to take them into account is to consider a nominal model affected by uncertainty. In addition, the measurement errors may vary due to change in external environments where the statistical properties of error in the system cannot be treated as unchanged, which thus results in filtering performance degradation or even divergence.

To fulfill the requirement of achieving the filter optimality or to preventing divergence problem of KF, the so-called adaptive Kalman filter (AKF) approach (Mehra, 1970, 1972; Mohamed and Schwarz, 1999; Hide et al., 2003) has been one of the promising strategies and has been widely explored for dynamically adjusting the parameters of the supposedly optimum filter based on the estimates of the unknown parameters for online estimation of motion as well as the signal and noise statistics available data. The AKF can be utilized as the noise-adaptive filter for tuning the noise covariance matrices. Adaptive filters are based on dynamically adjusting the parameters of the supposedly optimum filter based on the estimates of the unknown parameters. AKF can be based on an online estimation of motion as well as the signal and noise statistics available data.

In various circumstances where there are uncertainties in the system model and noise description, and the assumptions on the statistics of disturbances are violated due to the fact that in a number of practical situations, the availability of a precisely known model is unrealistic since in the modeling step, some phenomena are disregarded. The suboptimal configuration is typically based on a simplified error state dynamic/measurement model. One way to take them into account is to consider a nominal model affected by uncertainty. Many efforts have been made to improve the estimation of the covariance matrices based on the innovation-based estimation approach. The two major approaches that have been proposed for AKF are multiple model adaptive estimate and innovation adaptive estimation (IAE). The innovation sequences have been utilized by the correlation and covariance-matching techniques to estimate the noise covariances. The basic idea behind the covariance-matching approach is to make the actual value of the covariance of the residual consistent with its theoretical value. The IAE approach coupled with fuzzy logic techniques with membership functions designed using heuristic method has been very popular to adjust the noise statistics (Sasiadek et al., 2000).

The global positioning systems (GPS) (Farrell and Barth, 1999) are capable of providing accurate position information. While employed in the GPS receiver as the navigational state estimator, the EKF, the nonlinear version of KF, has been one of the promising approaches. In GPS navigation filter designs, there exist the model uncertainties which cannot be expressed by the linear state-space model. The linear model includes modeling errors since the actual vehicle motions are nonlinear process. The system model, system initial conditions, and noise characteristics have to be specified a priori. It is very often the case that little a priori knowledge is available concerning the maneuver.

Among various evolutionary optimizer techniques, genetic algorithms (GA) and particle swarm optimization (PSO) (Eberhart and Kennedy, 1995; Eberhart and Shi, 1998, 2001; Kennedy, 1997; Kennedy and Eberhart, 1995, 1997; Reynolds and Sverdlik, 1994) have attracted considerable attention. Recently, introduced by Kennedy and Eberhart (1995), PSO is a heuristic search method whose mechanics are inspired by swarming or collaborative behavior of biological populations. It is a population based stochastic searching technique and has been found to be robust and fast in solving nonlinear, non-differentiable, multimodal optimization problems. As an evolutionary optimization technique like the GA, PSO method mimics the development of this technique was inspired by the animal social behaviors such as school of fish, flock of birds, etc. and mimics the way they find food sources and prevent from predators. Unlike the GA, each particle in the PSO is assigned a random velocity, which determines the direction that the particle will fly through the search space. Since the PSO concept is relatively new, this has not been applied on wide range of practical applications as other evolutionary algorithms yet.

The method proposed in this paper makes use of the PSO techniques, which is employed into the GPS navigation filter for real-time identification of noise covariance matrices to prevent divergence of the KF. The remainder of this paper is organized as follows. In Section 2, the AKF algorithms are reviewed. In Section 3, the PSO algorithm is discussed. The proposed PSO-assisted AKF is presented in Section 4. Simulation results and discussion of the proposed method to GPS navigation processing is conducted in Section 5, where simulation results and discussion is provided. The conclusions are given in Section 6.

2. The adaptive extended Kalman filter

The EKF is a nonlinear version of Kalman filtering, which deals with the case governed by the nonlinear stochastic differential equations:
\[
\begin{align*}
\dot{x} &= f(x, t) + u(t) \quad (1a) \\
z &= h(x, t) + v(t) \quad (1b)
\end{align*}
\]

where the vectors \( u(t) \) and \( v(t) \) are both white noise sequences with zero means and mutually independent:
\[
E[u(t)u^T(\tau)] = Q \delta(t - \tau) \\
E[v(t)v^T(\tau)] = R \delta(t - \tau) \\
E[u(t)v^T(\tau)] = 0
\]

where \( \delta(t) \) is the Dirac delta function, \( E[\cdot] \) represents expectation, and superscript “T” denotes matrix transpose.

Expressing Equations (1a) and (1b) in discrete-time equivalent form leads to:
\[
\begin{align*}
x_{k+1} &= f(x_k, k) + w_k \quad (3a) \\
z_k &= h(x_k, k) + v_k \quad (3b)
\end{align*}
\]

where the state vector \( x_k \in \mathbb{R}^n \), process noise vector \( w_k \in \mathbb{R}^m \), measurement vector \( z_k \in \mathbb{R}^m \), measurement noise vector \( v_k \in \mathbb{R}^m \), \( \Phi_k \in \mathbb{R}^{m \times m} \), \( H_k \in \mathbb{R}^{m \times n} \), \( Q_k \) is the process noise covariance matrix and \( R_k \) is the measurement noise covariance matrix. In equation (3a) and (3b), both the vectors \( w_k \) and \( v_k \) are zero mean Gaussian white sequences having zero cross-correlation with each other:
\[
E[w_kw_k^T] = Q_k \delta_{kk}
\]
The discrete-time EKF algorithm is summarized as follow:

2.1 The extended Kalman filter
The discrete-time EKF algorithm is summarized as follow:
1. Initialize state vector and state covariance matrix: \( \hat{x}_0 \) and \( P_0 \).
2. Compute Kalman gain matrix from state covariance and estimated measurement covariance:
   \[
   K_k = P_k H_k^T \left( H_k P_k H_k^T + R_k \right)^{-1}
   \]
3. Multiply prediction error vector by Kalman gain matrix to get state correction vector and update state vector:
   \[
   \hat{x}_k = \hat{x}_k^- + K_k (z_k - \hat{z}_k), \text{ with } \hat{z}_k = h(\hat{x}_k, k)
   \]
4. Update error covariance:
   \[
   P_k = (I - K_k H_k) P_k^- 
   \]
5. Predict new state vector and state covariance matrix:
   \[
   \hat{x}_{k+1}^- = f(\hat{x}_k, k)
   \]
   \[
   P_{k+1}^- = \Phi_k P_k \Phi_k^T + Q_k
   \]
   where the linear approximation equations for system and measurement matrices are obtained through the relations:
   \[
   \Phi_k = \frac{\partial f_k}{\partial \hat{x}|_{x_k = \hat{x}_k}; \quad H_k = \frac{\partial h_k}{\partial \hat{x}|_{x_k = \hat{x}_k}}
   \]

Equations (7)–(9) are the measurement update equations, and equations (10)–(11) are the time update equations of the algorithm from \( k \) to \( k + 1 \). These equations incorporate a measurement value into a priori estimation to obtain an improved posteriori estimation. In the above equations, \( \Phi_k \) is the error covariance matrix defined by \( E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \), in which \( \hat{x}_k \) is an estimation of the system state vector \( x_k \), and the weighting matrix \( K_k \) is generally referred to as the Kalman gain matrix. The KF algorithm starts with an initial condition value, \( \hat{x}_0 \) and \( P_0 \). When new measurement \( z_k \) becomes available with the progression of time, the estimation of states and the corresponding error covariance would follow recursively ad infinity. Further detailed discussion on the EKF can be referred to Brown and Hwang (1997), Farrell and Barth (1999) and Gelb (1974).

2.2 Conventional adaptive extended Kalman filter
As discussed before, when implementing the KF, poor knowledge of the noise statistics may seriously degrade the KF performance, and even provoke the filter divergence. To fulfill the requirement, an AKF can be utilized as the noise-adaptive filter to estimate the noise covariance matrices. Mehra (1972) classified the adaptive approaches into four categories: Bayesian, maximum likelihood, correlation and covariance matching. The innovation sequences have been utilized by the correlation and covariance-matching techniques to estimate the noise covariances. The basic idea behind the covariance-matching approach is to make the actual value of the covariance of the residual consistent with its theoretical value. The implementation of IAE-based AKF to navigation designs has been widely explored. From the incoming measurement \( z_k \) and the optimal prediction \( \hat{x}_k^- \) obtained in the previous step, the innovations sequence is defined as:

\[
\nu_k = z_k - \hat{z}_k^- \]

The innovation reflects the discrepancy between the predicted measurement and the actual measurement. It represents the additional information available to the filter as a consequence of the new observation \( z_k \). The innovations sequence \( \nu_k \) is a zero-mean Gaussian white noise sequence. An innovation of zero means that the two are in complete agreement. The mean of the corresponding error of an unbiased estimator is zero. By taking variances on both sides, we have the theoretical covariance, the covariance matrix of the innovation sequence is given by:

\[
C_{\nu_k} = E[\nu_k \nu_k^T] = H_k P_k^- H_k^T + R_k
\]

which can be written as:

\[
C_{\nu_k} = H_k (\Phi_k P_k \Phi_k^T + Q_k) H_k^T + R_k
\]

The estimate of \( R_k \) can be performed by:

\[
\hat{R}_k = \hat{C}_{\nu_k} - H_k P_k^- H_k^T
\]

where \( \hat{C}_{\nu_k} \) is the statistical sample variance estimate of \( C_{\nu_k} \). Matrix \( \hat{C}_{\nu_k} \) can be computed through averaging inside a moving estimation window of size \( N \):

\[
\hat{C}_{\nu_k} = \frac{1}{N} \sum_{j=0}^{k} \nu_j \nu_j^T
\]

where \( N \) is the number of samples (usually referred to the window size); \( j_0 = k - N + 1 \) is the first sample inside the estimation window. Based on the residual based estimate, the estimate of process noise \( Q_k \) can be obtained:

\[
\hat{Q}_k = \frac{1}{N} \sum_{j=0}^{k} \Delta x_j \Delta x_j^T + P_k - \Phi_k P_{k-1} \Phi_k^T
\]

where \( \Delta x_j = x_j - \hat{x}_j^- \). This equation can be written in terms of the innovation sequence:

\[
\hat{Q}_k = K_k \hat{C}_{\nu_k} K_k^T
\]

If the window size \( N \) is too small, the estimation of measurement covariance will be too noisy. On the other hand, if a large window size is utilized, the estimation of measurement covariance will be smoother, however, at the expense of long transient time. The window size is chosen empirically (a reasonable size for the moving window may vary from 10 to 30) to give some statistical smoothing.
An AKF can be utilized as the noise-adaptive filter to estimate the noise covariance matrices and overcome the deficiency of KF. The benefit of the adaptive algorithm is that it keeps the covariance consistent with the real performance. The innovation sequences have been utilized by the correlation and covariance-matching techniques to estimate the noise covariances. The basic idea behind the covariance-matching approach is to make the actual value of the covariance of the residual consistent with its theoretical value. For more detailed information derivation for these equations (Mohamed and Schwarz, 1999).

One of the other approaches for adaptive processing is on the incorporation of fading factors. The idea of fading memory is to apply a factor matrix to the predicted covariance matrix to deliberately increase the variance of the predicted state vector:

$$P_{k+1} = \lambda_k P_k P_k^T + Q_k$$  \hspace{1cm} (19)

where $\lambda_k = \text{diag}(\lambda_1, \lambda_2, ..., \lambda_n)$. The main difference between different fading memory algorithms is on the calculation of scale factor matrix $\lambda_k$. One approach is to assign the scale factors as constants. When $\lambda_i \leq 1 (i = 1, 2, ..., n)$, the filtering is in a steady state processing while $\lambda_i > 1$, the filtering may tend to be unstable. For the case $\lambda_i = 1$, it deteriorates to the standard KF.

Another type of adaptation can be conducted by introducing a scaling factor directly to the $Q_k$ and/or $R_k$ matrices. To account for the greater uncertainty, the covariances need to be updated, through one of the following ways:

- $Q_k \leftarrow Q_{k-1} + \Delta Q_k$;
- $R_k \leftarrow R_{k-1} + \Delta R_k$.
- $Q_k \leftarrow Q_k \alpha^{-1}$;
- $R_k \leftarrow R_k \beta^{-1}$, $\alpha \geq 1$; $\beta \geq 1$.

For example, if equation (3a) and (3b) is utilized as an example, the filter equations can be augmented in the following way:

$$P_{k+1} = \Phi_k P_k \Phi_k^T + \alpha Q_k$$  \hspace{1cm} (20)

$$K_k = P_k H_k^T (H_k P_k H_k^T + \beta R_k)^{-1}$$

In case that $\alpha = \beta = 1$, it deteriorates to the standard KF.

### 3. The PSO algorithm

PSO is a population-based stochastic searching technique developed by Kennedy and Eberhart (1995). Among various evolutionary optimizer techniques, GA and PSO (Eberhart and Shi, 1998) have attracted considerable attention. The PSO is a robust stochastic evolutionary computation technique based on the movement and intelligence of swarms looking for the most fertile feeding location. Unlike the drawback of expensive computational cost of GA, PSO has better convergence speed. Similar to the GA, PSO is initialized with a population of random solutions. It starts with the random initialization of a population of individuals (particles) in the search space and works on the social behavior of the particles in the swarm. Unlike the GA, each particle is assigned a random velocity, which determines the direction that the particle will fly through the search space. Each particle keeps track of the coordinates associated with its best solution, which is calculated based upon a fitness function.

The concept of PSO is illustrated by the behaviors of a flock of birds or a school of fish. The algorithm maintains a swarm made up of a population of particles, where each particle represents a potential solution on an optimization problem. PSO tries to find the global best solution by simply adjusting the trajectory of each individual towards its own best location and towards the best particle of the swarm at each time step (also known as “generation”). The trajectory of each individual in the search space is adjusted by dynamically altering the velocity of each particle, according to its own flying experience and the flying experience of the other particles in the search space.

A swarm consists of a set of particles moving around the search space, each representing a potential solution (fitness). Each particle has a position vector $(x_k)$, a velocity vector $(v_k)$, the position at which the best fitness (PBest$_k$) encouheuristic search method whose mechanics are inspired by swarntered by the particle, and the index of the best particle (GBest) in the swarm. The position of each particle is updated every generation. This is done by adding the velocity to the position vector according to the following equation:

$$v_i = v_i + C_1 \times \text{rand}(\cdot) \times (\text{PBest}_i - x_i) + C_2 \times \text{rand}(\cdot) \times (\text{GBest} - x_i)$$  \hspace{1cm} (21)

At each iteration (generation), the position of each particle is updated based on their movement over a discrete time interval ($\Delta t$, which is usually set to 1) is according to the following equation:

$$x_i = x_i + v_i \times \Delta t$$  \hspace{1cm} (22)

The parameters $C_1$ and $C_2$ are set to positive constant values and are known as acceleration coefficients, which are normally taken as 2 whereas $\text{rand}(\cdot)$ represent uniformly distributed random values, uniformly distributed in $[0, 1]$ and $w$ is called as inertia weight, the inertia weight is employed to control the impact of the previous history of velocities on the current one. Accordingly, the parameter regulates the trade-off between the global and local exploration abilities of the swarm. A large inertia weight facilitates global exploration (searching new areas), while a small one tends to facilitate local exploration. Figure 1 shows the flow chart of the PSO algorithm.

### 4. The proposed PSO-assisted AKF

Innovation information is employed in the paper for assisting the design of PSO. The innovation information at the present epoch is employed for timely reflect the change in vehicle dynamics. To account for the greater uncertainty, Kalman filtering with motion detection can be incorporated into the GPS navigation filter for timely updated the covariance, through the relation given by equation (20), the new KF formulation is obtained. In case that $\alpha = 1$, it deteriorates to the standard KF. The innovation information at the recent epochs is collected for timely reflect the change in vehicle dynamic.

To detect the discrepancy between $\hat{C}_{av}$ and $C_{av}$, we define the degree of ratio of divergence (ROD) as the trace of innovation covariance matrix:

$$\text{ROD} = \frac{\text{Trace}(C_{av}) - \text{Trace}(\hat{C}_{av})}{\text{Trace}(C_{av})}$$
obtained, iteration is terminated and the actual one from sampled sequence. When the required FIT is so as to keeps the predicted covariance consistent with the navigation processing using the PSO-assisted AKF.

Figure 2 shows the flow chart for the PSO-assisted KF algorithm. The optimization problem is performed using PSO by iteratively tune the FIT parameter: $\text{FIT} = \frac{tr(\hat{C}_{k} \xi)}{tr[H_{k}(\Phi_{k}P_{k}\Phi_{k}^{T} + \alpha Q_{k})H_{k}^{T} + R_{k}]}$ (26)

so as to keeps the predicted covariance consistent with the actual one from sampled sequence. When the required FIT is obtained, iteration is terminated and the $\lambda_{k}$ is determined. Figure 2 shows the flow chart for the PSO-assisted KF algorithm. Figure 3 shows the configuration of the GPS navigation processing using the PSO-assisted AKF.

5. Simulation results and discussion

Simulation tests have been carried out to evaluate the GPS navigation performance for the proposed approach in comparison with the EKF approach. The computer codes were developed by the authors using the Matlab software. The commercial software Satellite Navigation (SATNAV) toolbox by GPSSoft LLC (2005) was employed for generating the satellite positions and pseudoranges.

The simulation scenario is as follows. The experiment was conducted on a simulated vehicle trajectory originating from the position of North 25.1492 degrees and East 121.7775 degrees at a constant altitude of 100 m. This is equivalent to $[-3,042,329.2, 4,911,080.2, 2,694,074.3]^T$ m in the WGS-84 ECEF coordinate system. The location of the origin is defined as the $(0, 0, 0)$ m location in the local tangent East-North-Up frame. The trajectory of the vehicle can be approximately divided into two categories according to the dynamic characteristics. The vehicle was simulated to conduct constant-velocity straight-line during the three time intervals, 0-250, 751-1,250 and 1,751-2,000 s, all at a constant speed of $10 \text{ m/s}$. Furthermore, it conducted circular motion with radius 2,500 m during 251-750 (counterclockwise), and 1,251-1,750 s (clockwise) where high dynamic maneuvering is involved.

Assume that the differential GPS (DGPS) mode is used and most of the errors can be corrected, but the multipath and receiver thermal noise cannot be eliminated. The dynamic process of the GPS receiver in medium dynamic environment can be represented by the position-velocity (PV) model.
In such case, the state to be estimated is a $8 \times 1$ vector, including three position components, three velocity components, and the receiver clock offset and drift errors, respectively. The states are implemented in the WGS-84 coordinates.

Since we assumed that the DGPS mode is used and most of the errors can be corrected, but the multipath and receiver measurement thermal noise cannot be eliminated. The measurement noise variances $r_i$ are assumed a priori known, which is set as 15 m$^2$. Let each of the white-noise spectral amplitudes that drive the random walk position states be $S_p = 0.003$ (m/s$^2$/rad/s). Also, let the clock model spectral amplitudes be $S_f = 0.4(10^{-18})$ s and $S_g = 1.58(10^{-18})$ s$^{-1}$. The measurement noise covariance matrix is given by:

$$R_k = \begin{bmatrix}
15 & 0 \\
15 & \ddots \\
0 & 15
\end{bmatrix}$$

Figure 4 shows the vehicle trajectory in simulation. The trajectory can be divided mainly into five intervals according their dynamic characteristics. The five segments which represent the trajectories in the five intervals are indicated on the figure. For clarity, the vehicle position in the east and north components is summarized in Table I, for providing better insight into vehicle dynamic information in each time interval. Figure 5 shows $t$ east and north components of the velocity for the simulated vehicle.

Figure 6 shows the flow chart of the PSO algorithm. Figure 7 shows GPS navigation solutions without and with adaptation.

![Figure 4](image1.png) **The vehicle trajectory in simulation**

![Figure 5](image2.png) **East and north components of the velocity for the simulated vehicle**

![Figure 6](image3.png) **Flow chart of the PSO algorithm**

![Figure 7](image4.png) **GPS navigation solutions without and with adaptation**

### Table I Description of vehicle motion in the five time intervals

<table>
<thead>
<tr>
<th>Segment</th>
<th>Time interval (s)</th>
<th>Motion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[0-250]</td>
<td>Constant velocity</td>
</tr>
<tr>
<td>2</td>
<td>[251-750]</td>
<td>Circular motion, counter-clockwise turn</td>
</tr>
<tr>
<td>3</td>
<td>[751-1,250]</td>
<td>Constant velocity</td>
</tr>
<tr>
<td>4</td>
<td>[1,251-1,750]</td>
<td>Circular motion, clockwise turn</td>
</tr>
<tr>
<td>5</td>
<td>[1,751-2,000]</td>
<td>Constant velocity</td>
</tr>
</tbody>
</table>
adaptation as compared to that by least-squares (LS) approach (as a baseline reference), where the parameters used are: ROD > 1.7, $|\eta| > 0.5$, window size $N = 5$. Figure 8 shows the east and north components of navigation errors and the $1\sigma$ bound using the proposed method (again, ROD > 1.7, $|\eta| > 0.5$). Comparison of ROD trajectories with and without PSO aiding is shown in Figure 9. In the two regions.

Figure 10 shows the history of convergence for the fitness function. When the fitness function approaches 1, then the Gbest is output, meaning that the optimization searching process has been completed. For better clarification, comparison of RMSE for the proposed approach as compared to the EKF and LS approaches is summarized in Table II. It can be seen that the noticeable improvement are in Segments 2 and 4, where the vehicle is conducting circular turns and is considered as the high dynamic regions.

It is seen that substantial improvement in tracking capability can be obtained, which confirms that the PSO has good potential in designing the AKF. Some principles for the setting of thresholds and window size are now discussed. Taking the limit for equation (24) leads to:

$$\lim_{N \to \infty} \left| \frac{1}{N} \sum_{j=0}^{k} (\hat{x}_k - \hat{x}_k) \right| = 0$$

(27)

The two parameters, window size $N$ and threshold $\eta$, have the relation:

$$N \propto \frac{1}{|\eta|}$$

(28)

Therefore, when the window size is set to a larger value, the threshold $\eta$ can be set to a smaller one. When the window size is increased, $\eta(\geq 0)$ is decreased toward 0, while ROD is increased toward 1, meaning that $tr(C_{nk})$ is approaching to $tr(C_{nk})$. For further illustration, examples are given in Figure 11, in which threshold $|\eta|$ and ROD trajectories are shown for window size 5 and 200, respectively. The value of window size has large influence on the response time, and accordingly the adaptation performance and navigation.
estimation accuracy. Trajectories of the $|\eta|$ and ROD for window size $N = 5$ and $N = 200$, respectively, are shown in Figures 12 and 13, respectively. It is seen that the transient time needed to reach convergence will increase if the window size is increased. Too large value for the window size is impractical in applications. Table III provides comparison of position errors for various combinations of ROD and $|\eta|$ with the same window size ($N = 5$); while Table IV provides comparison of position errors for various combinations ROD and $|\eta|$ with various window sizes ($N = 5, 10, 20, 30$).

Among the various combinations, the set of parameters: ROD = 1.7 and $|\eta|$ = 0.5 provides the best solutions for the example. However, the values are subject to change for the other cases. Therefore, further research can be conducted to find the best, ideal, or reasonable parameters for a specific application. As an example for illustrating the influence of the parameter setting to the estimation accuracy, Figure 14 is provided, which shows the east and north components of position errors and the $1 - \sigma$ bound using the proposed method with parameters ROD > 1 and $|\eta| > 0.1$. It is found that the performance in the three low dynamic regions (where the PV model matches the dynamic very well) has been degraded even that in the high dynamic regions have been improved.

6. Conclusions

This paper has presented a PSO assisted KF method for GPS navigation processing, where the conventional KF approach is coupled by the adaptive tuning system assisted by the PSO. The PSO provides the process noise covariance scaling factor for timely detecting the dynamical and environmental changes and implementing the online parameter tuning by monitoring the innovation information so as to maintain good tracking capability and estimation accuracy. The proposed method has the merits of good numerical stability since the matrices in the KF loop are able to remain positive definitive. Simulation experiments for GPS navigation have been provided to illustrate the accessibility. In addition, behavior of some innovation related parameters have been discussed, which are useful for providing the useful information in designing the AKF and for achieving the system integrity. The navigation accuracy based on the proposed method has been compared to the conventional EKF method and has demonstrated substantial improvement in both navigational accuracy and tracking capability.
Figure 13 Trajectories of the ROD for window size $N = 5$ (top) and $N = 200$ (bottom)

Table III Comparison of position errors for window size $N = 5$ and various combinations of ROD and $|\varphi|$:

<table>
<thead>
<tr>
<th>Segment</th>
<th>ROD &gt; 1.7</th>
<th>ROD &gt; 1.4</th>
<th>ROD &gt; 1.2</th>
<th>ROD &gt; 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$</td>
<td>\varphi</td>
<td>&gt; 0.5$</td>
<td>$</td>
</tr>
<tr>
<td>East component</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.4041</td>
<td>2.7991</td>
<td>3.0442</td>
<td>3.1685</td>
</tr>
<tr>
<td>2</td>
<td>3.7182</td>
<td>3.5806</td>
<td>3.4229</td>
<td>3.3201</td>
</tr>
<tr>
<td>3</td>
<td>1.3882</td>
<td>2.4501</td>
<td>2.7586</td>
<td>2.8351</td>
</tr>
<tr>
<td>4</td>
<td>3.1858</td>
<td>3.0454</td>
<td>3.0514</td>
<td>3.0789</td>
</tr>
<tr>
<td>5</td>
<td>1.3571</td>
<td>2.1851</td>
<td>2.5666</td>
<td>2.5705</td>
</tr>
<tr>
<td>North component</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.9926</td>
<td>2.3912</td>
<td>2.7007</td>
<td>2.7373</td>
</tr>
<tr>
<td>2</td>
<td>3.5052</td>
<td>3.1213</td>
<td>3.0610</td>
<td>3.0408</td>
</tr>
<tr>
<td>3</td>
<td>1.3323</td>
<td>2.1318</td>
<td>2.4433</td>
<td>2.5007</td>
</tr>
<tr>
<td>4</td>
<td>3.0811</td>
<td>2.8010</td>
<td>2.7501</td>
<td>2.7134</td>
</tr>
<tr>
<td>5</td>
<td>0.9793</td>
<td>1.4649</td>
<td>2.0717</td>
<td>2.1373</td>
</tr>
</tbody>
</table>

Note: Units: meters

References


Particle swarm optimization

Dah-Jing Jwo and Shun-Chieh Chang


About the authors

**Dah-Jing Jwo** received the PhD degree in aerospace engineering from the University of Texas at Arlington, Arlington, Texas, in 1995. From 1997 to 1998, he worked in industry and the Center for Aviation & Space Technology in the Industrial Technology Research Institute, Hsinchu, Taiwan. In 1998, he joined the Faculty of National Taiwan Ocean University, where he is currently a Professor and Chairman with the Department of Communications, Navigation and Control Engineering. His current research interests include GPS navigation, multisensor integrated navigation, estimation theory and applications, artificial intelligence, and vehicle guidance, navigation and control design. Dah-Jing Jwo is the corresponding author and can be contacted at: dijwo@mail.ntou.edu.tw

**Shun-Chieh Chang** received the BS degree in electronic engineering from Formosa University, Yunlin, in 2005, and the MS degree in communications, navigation and control engineering from National Taiwan Ocean University, Keelung, Taiwan, in 2007. His research interests include GPS navigation, artificial intelligence, and digital signal processing.

To purchase reprints of this article please e-mail: reprints@emeraldinsight.com
Or visit our web site for further details: www.emeraldinsight.com/reprints