

Research Article

A Fault-Tolerant Filtering Algorithm for SINS/DVL/MCP Integrated Navigation System

Xiaosu Xu,¹ Peijuan Li,² and Jian-juan Liu³

¹Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, School of Instrument Science & Engineering, Southeast University, Nanjing 210096, China

²Industrial Center, Nanjing Institute of Technology, Nanjing 211167, China

³Henan University of Technology, Zhengzhou 450007, China

Correspondence should be addressed to Xiaosu Xu; xxs@seu.edu.cn

Received 3 July 2014; Revised 13 April 2015; Accepted 15 April 2015

Academic Editor: Filippo Ubertini

Copyright © 2015 Xiaosu Xu 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.

The Kalman filter (KF), which recursively generates a relatively optimal estimate of underlying system state based upon a series of observed measurements, has been widely used in integrated navigation system. Due to its dependence on the accuracy of system model and reliability of observation data, the precision of KF will degrade or even diverge, when using inaccurate model or trustless data set. In this paper, a fault-tolerant adaptive Kalman filter (FTAKF) algorithm for the integrated navigation system composed of a strapdown inertial navigation system (SINS), a Doppler velocity log (DVL), and a magnetic compass (MCP) is proposed. The evolutionary artificial neural networks (EANN) are used in self-learning and training of the intelligent data fusion algorithm. The proposed algorithm can significantly outperform the traditional KF in providing estimation continuously with higher accuracy and smoothing the KF outputs when observation data are inaccurate or unavailable for a short period. The experiments of the prototype verify the effectiveness of the proposed method.

1. Introduction

Integration technology has become one of the key issues in navigation because of its capability of overcoming the drawbacks of single strapdown inertial navigation system (SINS), such as error accumulation and error swing over a period of time. The most commonly used navigation systems are radio and satellite based system, including global positioning system (GPS) [1] and Doppler velocity log (DVL) [2, 3]; their performance is fundamentally limited by propagation of signals [4, 5]. The integrated navigation system proposed in this work is an economic navigating algorithm of high precision, in which the SINS is integrated with a DVL and a magnetic compass (MCP).

The Kalman filter (KF) is an effective optimal estimation algorithm and has been widely used in integrated navigation system since 1970s [6–9]. On one hand, in order to achieve a better performance with a KF, an accurate system model

and reliable observation data are indispensable. Moreover, the KF fails to estimate the state vector exactly when the observations noise increases [10–12]. However, in real engineering applications, it is usually difficult to satisfy the above conditions. For example, undersea DVL's output will be interfered or broken owing to the cliffy sea bottom and other vessel reflections [13, 14]. In order to solve this problem, the following method is often adopted: if the observation data from DVL or MCP are temporarily inaccurate or inaccessible, the integrated navigation system can be compromised to work under pure inertial mode. But, it is evident that the measuring accuracy will deteriorate remarkably and the system error will accumulate or swing over time in pure inertial mode. After the system returns to the integration mode when the observation data are recovered, the position error will definitely not be compensated although the errors of velocities and attitude angles can be reduced. Therefore, a fault-tolerant filtering algorithm that can maintain

the accuracy of an integrated navigation system through inexact or unavailable observation data is important and urgent [15–17].

Evolutionary programming, put forward by Fogel [18], is essential for random searching in that it boosts the survival of the fittest and organic evolution mechanism and searches for the optimal point in feasible space via population search strategy [19, 20]. Moreover, the artificial neural network (ANN), optimized by evolutionary programming, not only overcomes the obstacle of conventional ANN which may easily plunge into local minimum and requires long training time, but also avoids the problem in genetic algorithms caused by binary-coded and cross operation. Thanks to its superiorities, ANN has been applied in many fields such as financial forecasting [21], breast cancer diagnosis [22], and electricity supply industry [23], and so forth [24].

In this paper, a fault-tolerant integrated navigation system with an adaptive Kalman filter (AKF) based on evolutionary artificial neural networks (EANN) is proposed, in which the EANN is used in self-learning and training of the intelligent data fusion algorithm. The proposed fault-tolerant adaptive Kalman filter (FTAKF) algorithm is able to maintain a smooth output under the condition when observation data are inaccurate or unavailable. The prototype experiment indicates that the algorithm can efficiently outperform the traditional KF with a higher accuracy when observation data are inaccurate or unavailable.

This paper is organized as follows. In Section 2, the inertial error model is introduced and a corresponding mathematical model in state and measurement space is described. In Section 3, we propose the AKF algorithm and compare its characteristics and properties with those of traditional KF algorithm. A fast ANN based on evolutionary programming is proposed in Section 4. In Sections 5 and 6, applications of algorithm in SINS/DVL/MCP are presented and several tests are conducted to evaluate the performance of the intelligent navigation system with FTAKF. Conclusions are made in Section 7.

2. Mathematical Model for Filter

2.1. Error Model of Integrated Navigation System

(1) *Inertial Error Model.* Here we define the navigation coordinates frame as an east-north-vertical (ENV) geography coordinates frame. The attitude angle, velocity, and position errors are given in the following. Attitude angle error is determined by

$$\dot{\boldsymbol{\phi}}^n = \delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n - (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \boldsymbol{\phi}^n + \mathbf{C}_b^n \boldsymbol{\varepsilon}^b, \quad (1)$$

where $\boldsymbol{\phi}^n = [\phi_E, \phi_N, \phi_U]^T$ is the orientation error vector of the calculated platform represented in the navigation coordinates frame, $\boldsymbol{\omega}_{ie}^n = [0, \omega_{ie} \cos L, \omega_{ie} \sin L]^T$ represents the rotation projection of the earth onto the ENV axes with $\omega_{ie} = 7.292115 \times 10^{-5}$ rad/s to be the angular velocity of the rotation

of the earth, and $\boldsymbol{\omega}_{en}^n = [-V_N/(R_n + h), V_E/(R_e + h), (V_E/(R_e + h)) \tan L]^T$ denotes the angular velocity of the rotation of a navigation coordinates frame relative to the earth with V_E , V_N , and V_U that represent the linear velocities, λ , L , and h that represent the position coordinates (longitude latitude and height, resp.), and R_n , R_e that represent the radii of the curvatures along the meridian and parallel, respectively (Table 2). Moreover, $\boldsymbol{\varepsilon}^b = [\varepsilon_x, \varepsilon_y, \varepsilon_z]$ is the gyro-drifts in the body coordinates frame with $\dot{\boldsymbol{\varepsilon}}^b = \mathbf{0}$. The last term \mathbf{C}_b^n represents the attitude matrix, which can be expressed by the attitude angles ψ , γ , and θ .

Given the definition $[s_\psi, c_\psi, s_\gamma, c_\gamma, s_\theta, c_\theta]^T = [\sin \psi, \cos \psi, \sin \gamma, \cos \gamma, \sin \theta, \cos \theta]^T$, \mathbf{C}_b^n can be written as

$$\mathbf{C}_b^n = \begin{bmatrix} c_\gamma c_\psi + s_\gamma s_\psi s_\theta & s_\psi c_\theta & s_\gamma c_\psi - c_\gamma s_\psi s_\theta \\ -c_\gamma s_\psi + s_\gamma c_\psi s_\theta & c_\psi c_\theta & -s_\gamma s_\psi - c_\gamma c_\psi s_\theta \\ -s_\gamma c_\theta & s_\theta & c_\gamma c_\theta \end{bmatrix}. \quad (2)$$

Velocity error is defined as

$$\begin{aligned} \delta\dot{\mathbf{V}}^n &= \mathbf{f}^n \times \boldsymbol{\phi}^n - (2\delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n) \times \mathbf{V}^n - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \\ &\quad \times \delta\mathbf{V}^n + \mathbf{C}_b^n \nabla^b, \end{aligned} \quad (3)$$

where $\delta\mathbf{V}^n = [\delta V_E, \delta V_N, \delta V_U]^T$ viewed as the linear velocity error vector is determined by the specific force vector $\mathbf{f}^n = [f_E, f_N, f_U]^T$ and the bias vector $\nabla^b = [\nabla_x, \nabla_y, \nabla_z]$ of accelerometer in the body coordinates frame, along with $\delta\boldsymbol{\omega}_{ie}^n$ and $\delta\boldsymbol{\omega}_{en}^n$ that represent the slow variations of $\boldsymbol{\omega}_{ie}^n$ and $\boldsymbol{\omega}_{en}^n$, respectively. Here, $\dot{\nabla}^b = \mathbf{0}$.

Position error is given by

$$\begin{aligned} \delta\dot{L} &= \frac{\delta V_N}{R_n + h} - \frac{V_N}{(R_n + h)^2} \delta h, \\ \delta\dot{\lambda} &= \frac{\delta V_E}{R_e + h} \sec L + \frac{V_E}{R_e + h} \sec L \tan L \delta L \\ &\quad - \frac{V_E}{(R_e + h)^2} \sec L \delta h, \\ \delta\dot{h} &= \delta V_U, \end{aligned} \quad (4)$$

where $\delta\lambda$, δL , and δh represent the position errors.

(2) *DVL Error Model.* The velocity of DVL in the body coordinates frame $\tilde{\mathbf{V}}_d$ can be described as follows:

$$\tilde{\mathbf{V}}_d = (\mathbf{I} + \delta\mathbf{K}) \mathbf{V}_d + \delta\mathbf{V}_d. \quad (5)$$

Here, $\delta\mathbf{K} = [\delta K_{dx}, \delta K_{dy}, \delta K_{dz}]^T$ is the scale factor error assumed to be random constant and $\delta\dot{\mathbf{K}} = \mathbf{0}$ and $\delta\mathbf{V}_d = [\delta V_{dx}, \delta V_{dy}, \delta V_{dz}]^T$ represents the linear velocity errors of

DVL, which can be characterized by first-order Markov process as

$$\begin{aligned}\delta\dot{V}_{dx} &= -\frac{1}{\tau_{dx}}\delta V_{dx} + w_{dx}, \\ \delta\dot{V}_{dy} &= -\frac{1}{\tau_{dy}}\delta V_{dy} + w_{dy}, \\ \delta\dot{V}_{dz} &= -\frac{1}{\tau_{dz}}\delta V_{dz} + w_{dz},\end{aligned}\quad (6)$$

where $\boldsymbol{\tau}_d = [\tau_{dx}, \tau_{dy}, \tau_{dz}]^T$ is the time correlation coefficient of this Markov process and $\mathbf{w}_d = [w_{dx}, w_{dy}, w_{dz}]^T$ is the input noise of DVL.

(3) *MCP Error Model*. Heading attitude angle error of MCP $\delta\psi_m$ can be represented as first-order Markov process with

$$\delta\dot{\psi}_m = -\frac{1}{\tau_m}\delta\psi_m + w_m, \quad (7)$$

where τ_m is the time correlation coefficient and w_m is the input noise of MCP.

2.2. *State Space*. The error models given in (1), (3)-(4), and (6)-(7) have a standard form given by

$$\dot{\mathbf{X}} = \mathbf{F} \cdot \mathbf{X} + \mathbf{\Gamma} \cdot \mathbf{W}, \quad (8)$$

where matrices \mathbf{F} and $\mathbf{\Gamma}$ can be established from the above error model and \mathbf{W} is the input (system) noise:

$$\begin{aligned}\mathbf{W} \\ = [w_{gx}, w_{gy}, w_{gz}, w_{ax}, w_{ay}, w_{az}, w_{dx}, w_{dy}, w_{dz}, w_m],\end{aligned}\quad (9)$$

where w_{gx} , w_{gy} , and w_{gz} are the white noise of the gyros and w_{ax} , w_{ay} , and w_{az} are the white noise of the accelerometers. The state vector \mathbf{X} of the integrated navigation system is defined as

$$\begin{aligned}\mathbf{X} = [\delta V_E, \delta V_N, \delta V_U, \phi_E, \phi_N, \phi_U, \delta L, \delta \lambda, \delta h, \nabla_x, \nabla_y, \\ \nabla_z, \varepsilon_x, \varepsilon_y, \varepsilon_z, \delta V_{dx}, \delta V_{dy}, \delta V_{dz}, \delta K_{dx}, \delta K_{dy}, \delta K_{dz}, \\ \delta\psi_m].\end{aligned}\quad (10)$$

2.3. *Measurement Space*. Supposing that the error of the velocity of DVL in the navigation coordinates frame \mathbf{V}_d^n is mainly caused by platform misalignment, the error of $\tilde{\mathbf{V}}_d^n$ can be described as

$$\begin{aligned}\tilde{\mathbf{V}}_d^n &= \mathbf{V}_d^n + \delta\mathbf{V}_d^n = \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \delta\mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d \\ &= \mathbf{V}_d^n + \delta\mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d,\end{aligned}\quad (11)$$

where \mathbf{C}_d^b is a constant matrix which translates the velocity of DVL \mathbf{V}_d from the instrumental frame to the body coordinates

frame. The perturbation of the attitude matrix $\delta\mathbf{C}_b^n$ is defined as

$$\delta\mathbf{C}_b^n = (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n, \quad (12)$$

where p is the calculating navigation coordinate frame. Substituting (12) into (11), it yields

$$\tilde{\mathbf{V}}_d^n = \mathbf{V}_d^n + (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d. \quad (13)$$

After differentiating the velocity of SINS and DVL, the measurement model can be written as

$$\begin{aligned}\Delta\mathbf{V}^n &= \tilde{\mathbf{V}}^n - \tilde{\mathbf{V}}_d^n + \boldsymbol{\eta}_d \\ &= \mathbf{V}^n + \delta\mathbf{V}^n - \mathbf{V}_d^n - (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \boldsymbol{\eta}_d \\ &= \delta\mathbf{V}^n - (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \boldsymbol{\eta}_d,\end{aligned}\quad (14)$$

where the measurement velocity vector $\Delta\mathbf{V}^n = [\Delta V_E, \Delta V_N, \Delta V_U]$ and $\delta\mathbf{V}_d^n = (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d$ can be calculated from

$$\begin{aligned}\delta V_{dE} &= \phi_U V_N - \phi_N V_U + \mathbf{C}_b^n(1, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(1, 2) \delta V_{dy} + \mathbf{C}_b^n(1, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(1, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(1, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(1, 3) V_U \delta K_{dz}, \\ \delta V_{dN} &= \phi_E V_U - \phi_U V_E + \mathbf{C}_b^n(2, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(2, 2) \delta V_{dy} + \mathbf{C}_b^n(2, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(2, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(2, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(2, 3) V_U \delta K_{dz}, \\ \delta V_{dU} &= \phi_N V_E - \phi_E V_N + \mathbf{C}_b^n(3, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(3, 2) \delta V_{dy} + \mathbf{C}_b^n(3, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(3, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(3, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(3, 3) V_U \delta K_{dz}.\end{aligned}\quad (15)$$

Similarly, after differentiating the heading angle of SINS and MCP, the measurement model is given as

$$\begin{aligned}\Delta\psi &= \tilde{\psi} - \tilde{\psi}_m + \eta_m = \psi + \delta\psi - \psi_m - \delta\psi_m + \eta_m \\ &= \delta\psi - \delta\psi_m + \eta_m,\end{aligned}\quad (16)$$

where

$$\begin{aligned}\delta\psi &= -\frac{\mathbf{C}_b^n(1, 2) \cdot \mathbf{C}_b^n(3, 2)}{[\mathbf{C}_b^n(1, 2)]^2 + [\mathbf{C}_b^n(2, 2)]^2} \phi_E \\ &\quad - \frac{\mathbf{C}_b^n(2, 2) \cdot \mathbf{C}_b^n(3, 2)}{[\mathbf{C}_b^n(1, 2)]^2 + [\mathbf{C}_b^n(2, 2)]^2} \phi_N + \phi_U.\end{aligned}\quad (17)$$

Based on the previous analysis, the measurement model is defined as

$$\mathbf{Z}_k = \mathbf{H}_k \mathbf{X}_k + \boldsymbol{\eta}_k, \quad (18)$$

where $\mathbf{Z} = [\Delta V_E, \Delta V_N, \Delta V_U, \Delta \psi]^T$ is the observation vector, \mathbf{H}_k is the observation matrix at the moment of k which can be established from (15) and (17), and vector $\boldsymbol{\eta}_k$ represents the Gaussian measurement noise with zero mean value and covariance matrix \mathbf{R} .

3. AKF Algorithm to Integrated Navigation System

Given the differential equations, we can have a continuous state transition matrix \mathbf{F} for error states. The continuous transition matrix can be discretized for KF application by using the first-order Taylor series approximation. The corresponding discrete transition matrix can be computed as

$$\boldsymbol{\Phi} = \mathbf{I} + \mathbf{F}\Delta T, \quad (19)$$

where ΔT is chosen as 0.01 s. Thus, the state equations and observation equation can be written as

$$\begin{aligned} \mathbf{X}_k &= \boldsymbol{\Phi}_{k,k-1} \mathbf{X}_{k-1} + \boldsymbol{\Gamma}_{k-1} \mathbf{W}_{k-1}, \\ \mathbf{Z}_k &= \mathbf{H}_k \mathbf{X}_k + \boldsymbol{\eta}_k, \end{aligned} \quad (20)$$

where \mathbf{X}_k is the 22-dimensional state vector at the moment of k , \mathbf{Z}_k is the 4-dimensional observation vector at the moment of k , $\boldsymbol{\Phi}_{k,k-1}$ is state-transfer-matrix from sample time $k-1$ to k , $\boldsymbol{\Gamma}_{k-1}$ is the system noise matrix at the moment of $k-1$, and \mathbf{W}_{k-1} is system noise. Here, $\{\mathbf{W}_k\}$ and $\{\boldsymbol{\eta}_k\}$ are uncorrelated white noise sequences with zero mean. So we can have $E\{\mathbf{W}_k \mathbf{W}_j^T\} = \mathbf{Q}_k \delta_{kj}$ and $E\{\boldsymbol{\eta}_k \boldsymbol{\eta}_j^T\} = \mathbf{R}_k \delta_{kj}$. \mathbf{Q}_k is the variance matrix of system noise and \mathbf{R}_k is the variance matrix of observation noise.

The equations of AKF based on the new information sequence are written as

$$\begin{aligned} \widehat{\mathbf{X}}_{k/k-1} &= \boldsymbol{\phi}_{k,k-1} \widehat{\mathbf{X}}_{k-1}, \\ \mathbf{V}_k &= \mathbf{Z}_k - \mathbf{H}_k \widehat{\mathbf{X}}_{k/k-1}, \\ \mathbf{P}_{k/k-1} &= \mathbf{K}_{k-1} \mathbf{V}_k \mathbf{V}_k^T \mathbf{K}_{k-1}^T + \mathbf{P}_{k-1}, \\ \mathbf{K}_k &= \mathbf{P}_{k/k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \\ \widehat{\mathbf{X}}_k &= \widehat{\mathbf{X}}_{k/k-1} + \mathbf{K}_k \mathbf{V}_k, \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k/k-1}. \end{aligned} \quad (21)$$

Compared with traditional KF equations, there are two differences. Firstly, \mathbf{Q}_k does not appear in above filter equations. That is to say, the statistical properties of the system noise are not needed in AKF, which is a highlight of this algorithm. Secondly, the initial value \mathbf{K}_0 is needed, and it can be obtained by applying the traditional KF in the first loop. The AKF scheme of the integrated navigation system is shown in Figure 1.

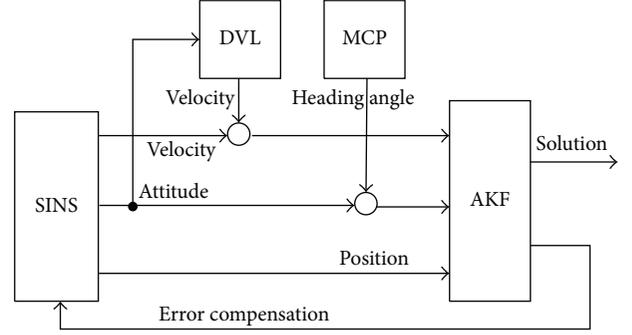


FIGURE 1: AKF scheme of SINS/DVL/MCP.

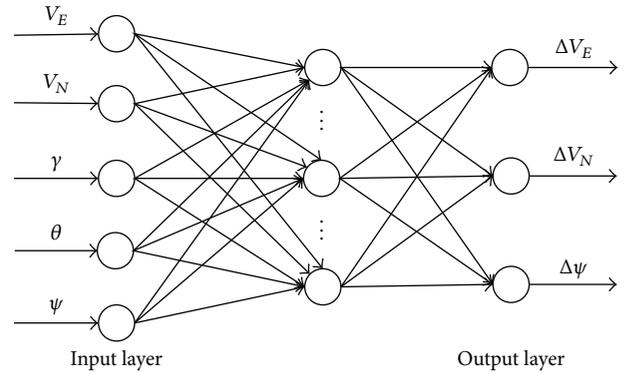


FIGURE 2: Artificial neural networks model.

4. Evolutionary Artificial Neural Networks

In recent years, ANN has been widely used in various fields, especially the backpropagation (BP) training [25]. However, drawbacks like poor convergence and failure in obtaining the global optimum are the bottleneck in the practical applications [26]. Articles [27–29] use genetic algorithms to optimize the parameters of neural networks. But there exist some imperfections. In this work, evolutionary programming is used to optimize the structure and connection weights in neural networks.

4.1. Artificial Neural Networks Model. ANN theory has proved that three-layer feedforward networks containing only one hidden layer can be a nonlinear mapping of any system with any precision [30]. According to this, the ANN model used in this paper consists of an input layer, a hidden layer, and an output layer. The network structure shown in Figure 2 treats the output navigation parameters of SINS as the input of the networks and the measurement errors as the output of the networks. Considering the limitation of underwater experimental environment, tests are conducted on a land vehicle, and the experimental area is flat and the speed of the vertical direction V_U can be regarded as zero. Hence, there are 5 nodes in input layer while there are 3 in output layer.

Based on the assumption that the number of hidden layer nodes is m , we have $\mathbf{x}' \in \mathbf{R}^m$, $\mathbf{x}' = (x'_0, x'_1, \dots, x'_{m-1})^T$, and the connection weights between input and hidden layers are $\hat{\omega}_{ij}$ ($i = 0, 1, \dots, 4$; $j = 0, 1, \dots, m-1$), with thresholds to be Θ_j . The connection weights between output and hidden layers are represented as $\hat{\omega}_{jk}$ ($k = 0, 1, 2$), with thresholds to be Θ_k . Then the output of each layer can be written as

$$\begin{aligned} x'_j &= f_1 \left(\sum_{i=0}^4 \hat{\omega}_{ij} x_i - \Theta_j \right), \\ y_k &= f_2 \left(\sum_{j=0}^{m-1} \hat{\omega}_{jk} x'_j - \Theta_k \right), \end{aligned} \quad (22)$$

where f_1 and f_2 are transfer functions.

In order to study the nonlinear properties of biological neurons, usually f_1 is chosen as a sigmoid function $f_1(x) = 1/(1 + e^{-x})$ and f_2 is chosen as a linear function $f_2(x) = x$. During optimizing, the threshold value is treated as import power with value of -1 .

4.2. Artificial Neural Networks Optimized by Evolutionary Programming. Darwin's evolutionary programming is a new search algorithm which is largely dependent on the mutation operator, while the genetic algorithms rely greatly on the crossover operator. When using the evolutionary programming as an optimization algorithm, users need not to be constrained to a particular code structure (representation) or a mutation strategy (evolution operator). Therefore, the flexibility and convenience of evolutionary programming make it surpass the conventional genetic algorithm. The steps for ANN optimization with the network structure and parameters using evolutionary programming are described as follows.

(1) *Initialization.* Here the population of neural networks is set by an initial quantity and the number of hidden layer neurons and the connection weights are initialized randomly within a certain range. Population can be expressed as $\mathbf{P} = \{X_1, X_2, \dots, X_{2N}\}$, where X_i ($i = 1, 2, \dots, 2N$) is the weight distribution of the network and defined as $X_i = [W_1, W_2, \dots, W_M]$ with W_k ($k = 1, 2, \dots, M$) to be the weight between every two connecting nodes of the network.

(2) *Performance Calculation and Categorization.* The training procedure of neural networks uses the overall mean square deviation as a metric. Training will be terminated if the networks' overall mean square deviation is less than a preset value. Here, the set error is defined as the individual survival environment, which is the goal of every individual evolutionary. For each individual, the fitness function is defined as

$$\begin{aligned} f &= \frac{1}{1 + E}, \\ E &= \frac{1}{2} \sum_{l=1}^p \sum_{k=1}^3 (d_k - o_k)^2, \end{aligned} \quad (23)$$

where p is the number of the imported samples and d_k and o_k are the desired and actual outputs of the node k of the output layer. Each individual is sorted in a descending order according to fitness.

(3) *Termination Condition Check.* When the best individual meets the requirement or number of loops exceeds the limitation of evolutionary generation, turn to *Step (5)*; otherwise continue.

(4) *New Population Generation.* Firstly, n individuals in the current generation are picked as part of the next generation, while removing the rest. Secondly, n new individuals are generated from individuals selected by mutation. These $2n$ individuals gather to form the new population. Here Gaussian variation is used as the mutation operator. Moreover, according to the characteristics of BP neural networks, the actual error of network is introduced as the enlightening knowledge in the formula; namely,

$$\begin{aligned} W_k(t+1) &= W_k(t) + \Delta W_k, \\ \Delta W_k &= L \sqrt{\frac{1}{F_{X_i}}} N(0, 1), \\ L &= C \exp \frac{n}{n_{\max}}, \end{aligned} \quad (24)$$

where $W_k(t+1)$ and $W_k(t)$ are the weights of the $(t+1)$ th generation and (t) th generation, respectively, with subscript to be $k = 1, 2, \dots, M$, $N(0, 1)$ represents a standard normal distribution, L and C are definition coefficients, n_{\max} is the maximum training cycles of the network, and n is the number of the individual survival generations. When the new population is generated, jump to *Step (2)*.

(5) *Termination Cycle.* Terminate the evolution procedure and then train the optimal neural networks using BP algorithm until all requirements are met.

5. Applications of Algorithm in SINS/DVL/MCP

On account of the previous algorithm, the FTAKF method based on EANN technology can be applied to the SINS/DVL/MCP integrated system.

The integrated navigation system with an EANN based AKF is shown in Figure 3.

When the observation data are diagnosed as available and reliable, EANN works at training state and uses SINS velocity and attitude information as input while the measurement errors are treated as the desired output. In this case switch 1 is connected and switch 2 is off. When the observation data is unavailable or unreliable, EANN works at forecasting state, with switch 1 to be off and switch 2 to be connected.

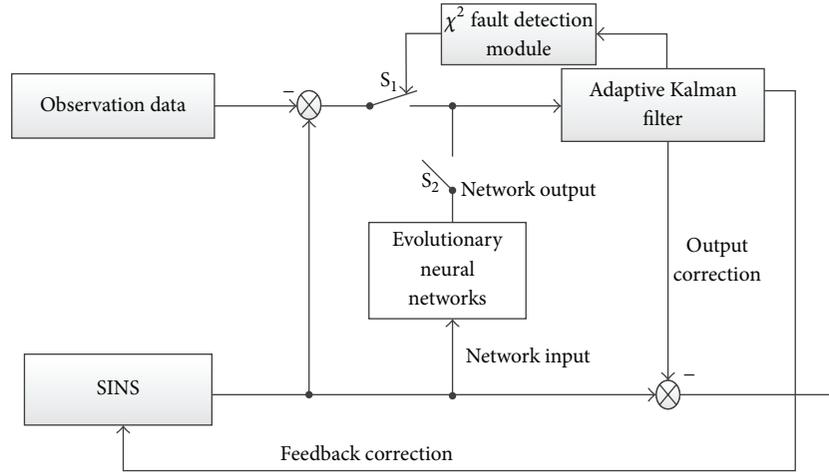


FIGURE 3: Integrated navigation system blocks diagram based on evolutionary neural networks.

Here, we use χ^2 detecting method to judge whether the observation data is available [31].

When no fault occurs, the residual of filter belongs to white noise, where

$$\mathbf{V}_k \sim N(\mathbf{0}, \mathbf{A}_k). \quad (25)$$

The mean of \mathbf{V}_k is $\mathbf{0}$ and its variance is

$$\mathbf{A}_k = \mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{R}_k. \quad (26)$$

While the auxiliary navigation equipment fails, the statistical characteristic of measurements noise changes. Then, the residual \mathbf{V}_k no longer belongs to the white noise. So the mean of the residual \mathbf{V}_k does not equal $\mathbf{0}$, which can be used to judge whether the observation data is available.

Firstly, we define the detection function as

$$\gamma_k = \mathbf{V}_k^T \mathbf{A}_k^{-1} \mathbf{V}_k, \quad (27)$$

where γ_k obeys the χ^2 distribution and its degree of freedom is m ; that is, $\gamma_k \sim \chi^2(m)$, and m is the dimension of observations vector \mathbf{Z}_k .

Then, the detection criteria are as follows:

$$\begin{aligned} &\text{if } \gamma_k \geq T_D, \text{ observation data are unavailable,} \\ &\text{if } \gamma_k < T_D, \text{ observation data are available,} \end{aligned} \quad (28)$$

where the threshold T_D determines the performance of the detection method, which should consider the leakage alarm rate and the false alarm rate.

From (25), (26), and (27), we can conclude that the fault information needs to be great enough; then the value of the detection function γ_k will be bigger than T_D . So this method is suitable for the mutant fault detection. The noise sudden increasing fault and gradual fault detection method will be specially introduced in another manuscript of the authors.

TABLE 1: IMU parameters of tested system.

Terms	Parameters
Accelerometer biases (g)	$< 80 \times 10^{-5}$
Accelerometer nonlinearity (% FS)	$< 30 \times 10^{-3}$
Accelerometer resolution (mg)	0.5
Gyro biases ($^{\circ}/\text{hr}$)	< 0.1
Gyro nonlinearity (% FS)	$< 20 \times 10^{-2}$
Random walk ($^{\circ}/\text{hr}^{1/2}$)	< 0.1

6. Experiments

Limited by the experimental condition, on-board experiments were carried out to instead of underwater tests. The defect of the on-board experiments was that the movement of autonomous underwater vehicle (AUV) in the direction of depth cannot be simulated, but the other conditions are similar.

(1) *Experiment Equipment.* To evaluate the performance of the proposed fault-tolerant filtering algorithm, several tests are conducted on a land vehicle platform equipped with integrated navigation system (tested system) and PHINS III (comparison unit).

- (i) *Tested System: Integrated Navigation System (Strap-down Inertial Navigation System, Speed Log, and Magnetic Compass).* The inertial measurement unit (IMU) parameters of the tested system are shown in Table 1.
- (ii) *Comparison Unit.* PHINS III is produced by IXSEA cooperation which worked on SINS/GPS mode.
- (iii) *Vehicle Test Environment.* The vehicle test structure is shown as Figure 4. For the outputs of PHINS are used as the benchmark, the IMU which we used is fixed on a platform with the PHINS. The installation diagram is as Figure 5. The working scene of vehicle test inside the car is shown in Figure 6.

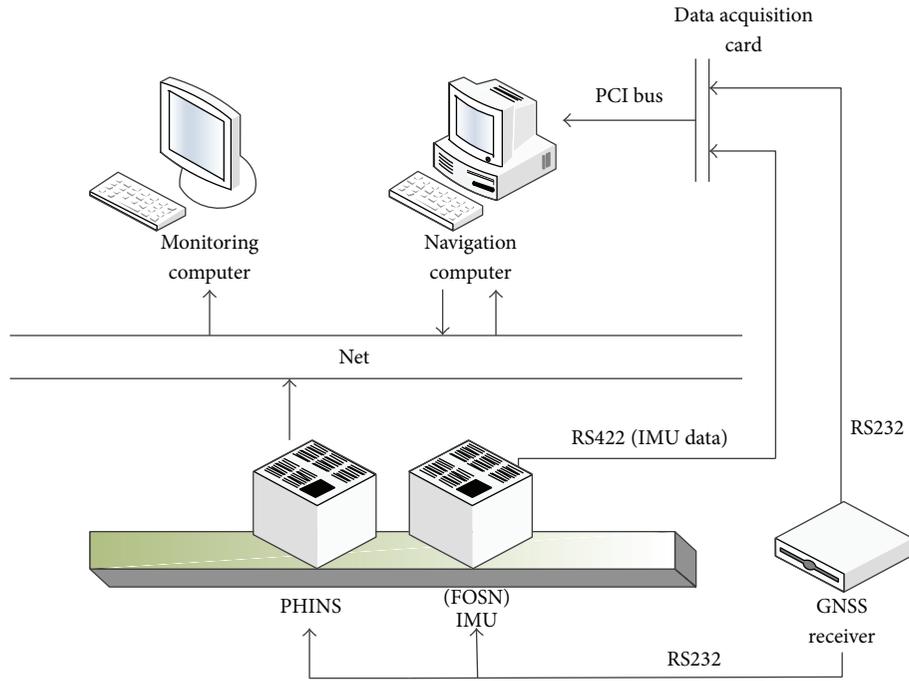


FIGURE 4: Vehicle test structure.



FIGURE 5: Experimental equipment: tested system and comparison unit.



FIGURE 6: Working scene.

(2) *Experiment Route.* The experiment was conducted on the campus of Southeast University.

TABLE 2: Statistical results of horizontal position errors.

	Longitude error (unit: m)	Latitude error (unit: m)
Experiment 1	141	798
Experiment 2	1.5	21

Experiment 1 (for a traditional KF algorithm). The experiment consists of three phases.

Phase 1. From start time as the yellow arrow on the map to 1000 seconds, this phase marked by yellow line on the map, the tested system works in the integration mode, and the observation data are available.

Phase 2. From 1000 seconds to 1600 seconds marked by black line on the map, the tested system works in the pure inertial mode, and the observation data are unavailable.

Phase 3. From 1600 seconds to 2800 seconds marked by light green line on the map, the tested system recovers from the pure inertial mode and works in the integration mode, while the observation data are renewed again.

Figures 7–9 show the system’s navigation errors of Experiment 1. Figure 10 shows the navigation route of Experiment 1.

According to Figures 7 to 9, during Phase 2, the horizontal attitude errors are bigger than the attitude errors when the system is working in the integrated mode. For lack of outside velocity and yaw information, the yaw error, velocity error, and the position errors are increased fast as time grows when

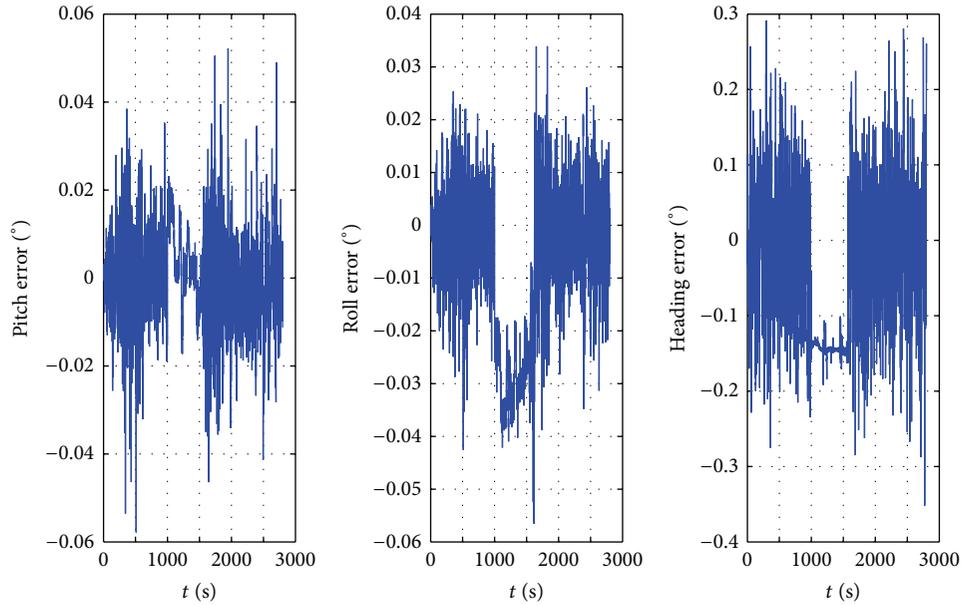


FIGURE 7: Experiment 1, attitude errors.

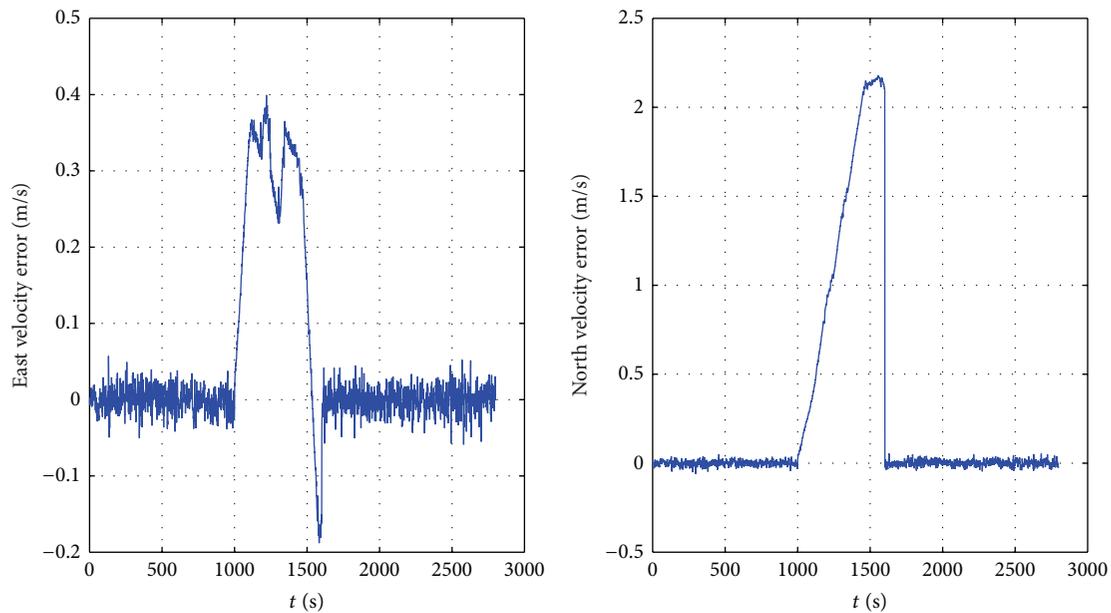


FIGURE 8: Experiment 1, velocity errors.

the system works in the pure inertial mode, and the precision of the system depends highly on IMU's accuracies.

From 1600 seconds, the observation data are available again and the system returns to the integration mode. The attitude and velocity errors can be lessened due to correction effects of KF, while the position errors cannot be eliminated.

(3) Experiment Results

Experiment 2 (for the FTAKF algorithm based on EANN). The experiment also consists of three phases like Experiment 1. In order to verify the effects of evolutionary

neural networks, the observation data will be unavailable during Phase 2 from 1000 to 1600 seconds; the tested system will work in the fault-tolerant filtering mode instead of the pure inertial mode.

Experiment results shown in Figures 11–13 indicate that when the observation data are unavailable, the evolutionary neural network, which has completed training process, works on the forecasting mode and provides KF with required inputs. Thus, the KF can keep operating in a continuous manner and the system precision can be maintained stably. Figure 14 shows the navigation route of Experiment 2.

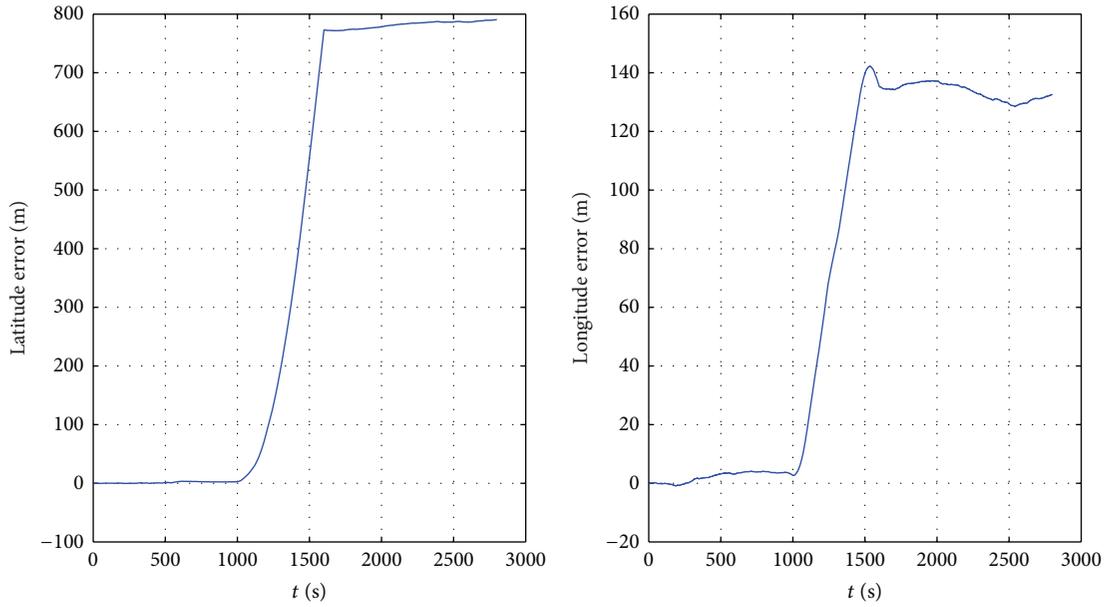
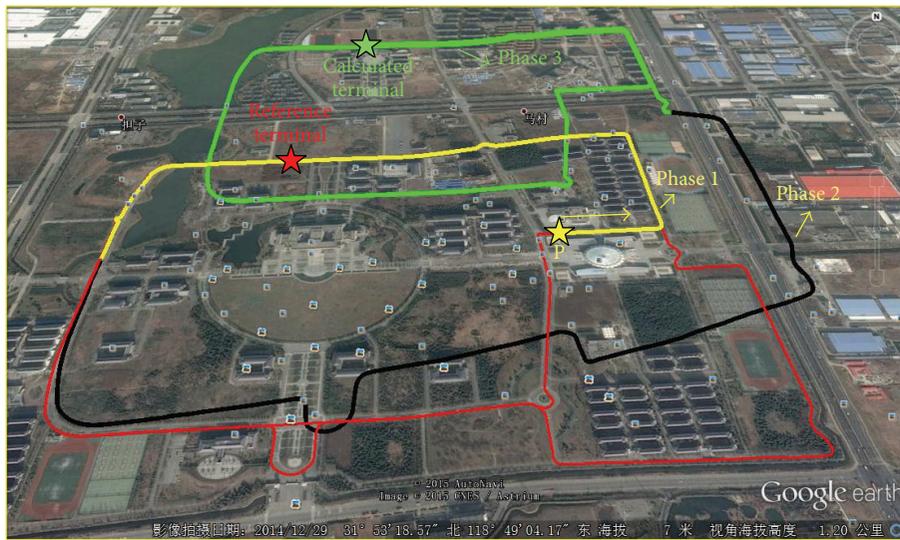


FIGURE 9: Experiment 1, position errors.



- Phase 1
- Phase 2
- Phase 3
- Reference route
- ★ Start point
- ★ Calculated terminal
- ★ Reference terminal

FIGURE 10: Experimental route map with traditional KF algorithm.

7. Conclusion

In this paper, aimed at addressing the problems of inaccurate model and unreliable observation data for KF in practical applications of AUV (autonomous underwater vehicle), a FTAKF algorithm for integrated navigation system based on EANN is proposed and tested. The conclusions are listed as follows:

(1) The performance of a traditional KF is vulnerable due to its dependence on precision of the system model,

such as state equation, observation equation, and noise statistic characters. In order to reduce the dependence on the precision of the system model, an AKF algorithm is designed.

(2) The FTAKF algorithm based on EANN is capable of maintaining a smooth filter output even under the severe condition when observation data are inaccurate or unavailable and of enhancing the precision and reliability of integrated navigation system.

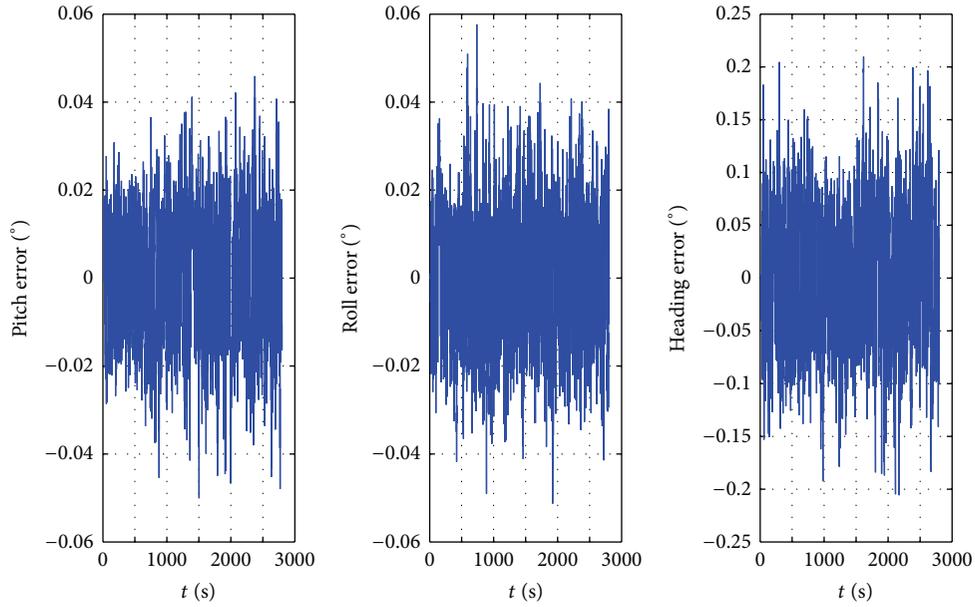


FIGURE 11: Experiment 2, attitude errors.

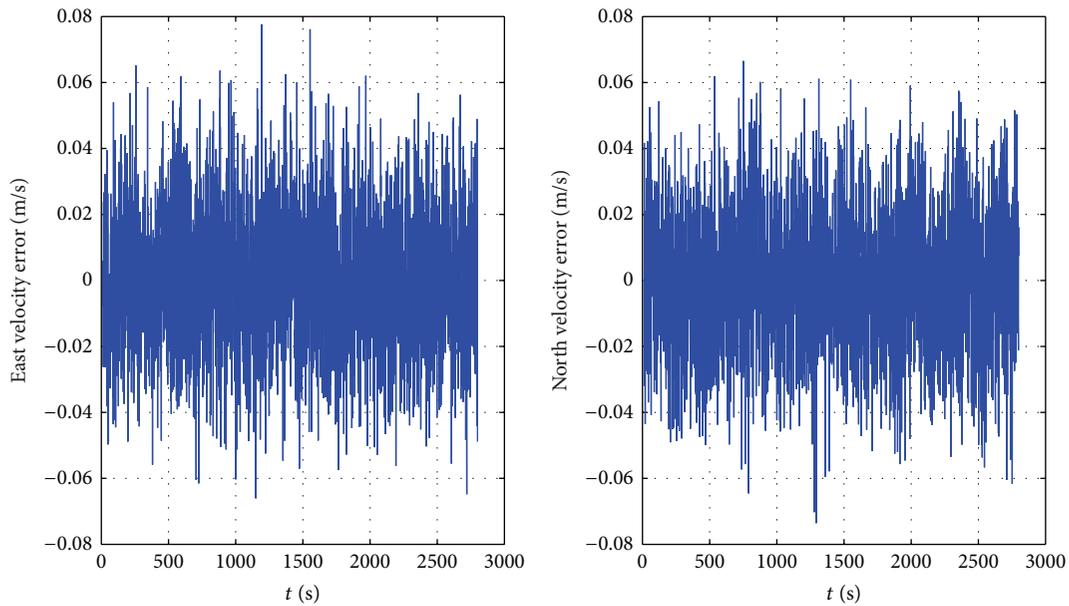


FIGURE 12: Experiment 2, velocity errors.

- (3) The results of experiments show that the FTAKF algorithm based on EANN proposed in this paper is effective and practical for engineering application.
- (4) It should be noted that, because of the utilization of the evolutionary neural network, the prerequisite for this system is that the training time of neural networks must be longer than the period that observation data is unavailable.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the Chinese National Natural Science Foundation (51175082, 60874092) and the Project

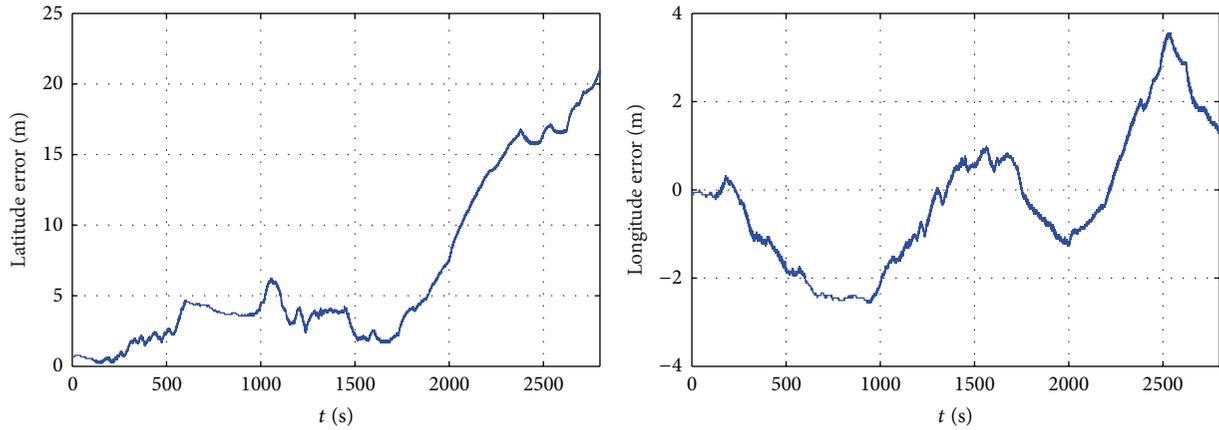


FIGURE 13: Experiment 2, position errors.

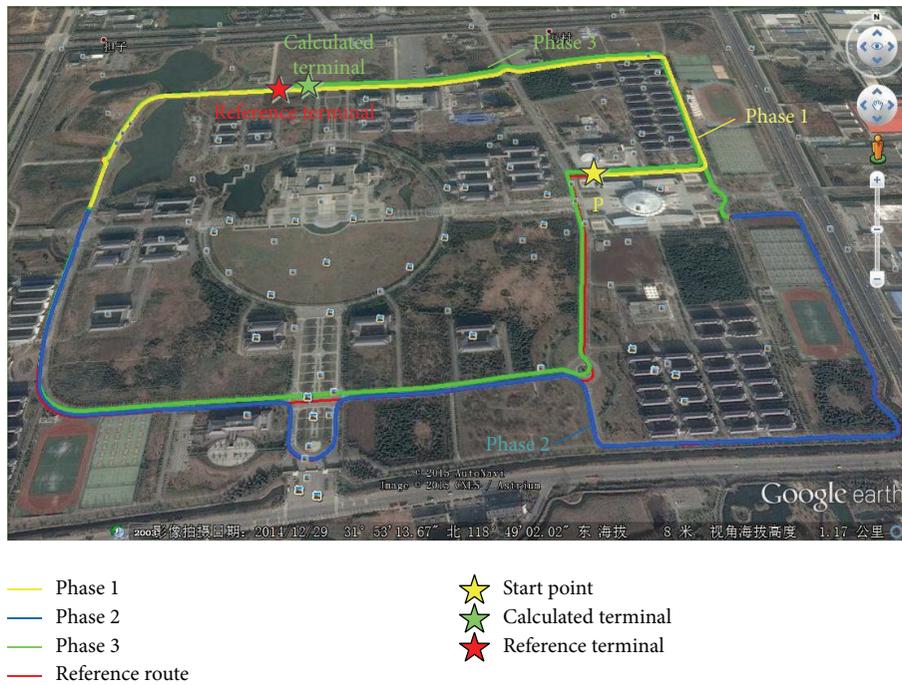


FIGURE 14: Experimental route map with FTAKF algorithm.

Supported by the Foundation of Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, China (201404), for the support of this work.

References

[1] R. McEwen, H. Thomas, D. Weber, and F. Psota, "Performance of an AUV navigation system at arctic latitudes," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 443–454, 2005.

[2] P. Ridao, D. Ribas, E. Hernandez, and A. Rusu, "USBL/DVL navigation through delayed position fixes," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 2344–2349, Shanghai, China, May 2011.

[3] B. Jalving, K. Gade, K. Svartveit, A. Willumsen, and R. Sørhagen, "DVL velocity aiding in the HUGIN 1000 integrated inertial navigation system," *Modeling, Identification and Control*, vol. 25, no. 4, pp. 223–235, 2004.

[4] T. Zhang and X. Xu, "A new method of seamless land navigation for GPS/INS integrated system," *Measurement*, vol. 45, no. 4, pp. 691–701, 2012.

[5] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: a review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.

[6] M. J. Dove, "Kalman filter techniques in marine integrated navigation systems," *Journal of Navigation*, vol. 30, no. 1, pp. 135–145, 1977.

[7] S. A. Broatch and A. J. Henley, "An integrated navigation system manager using federated Kalman filtering," in *Proceedings of the*

- IEEE National Aerospace and Electronics Conference (NAECON '91)*, pp. 422–426, Dayton, Ohio, USA, May 1991.
- [8] X.-Y. Chen, J. Yu, and X.-F. Zhu, “Theoretical analysis and application of Kalman filters for ultra-tight global position system/inertial navigation system integration,” *Transactions of the Institute of Measurement and Control*, vol. 34, no. 5, pp. 648–662, 2012.
- [9] T. Xu, “Adaptive Kalman filter for INS/GPS integrated navigation system,” in *Proceedings of the 2nd International Conference on Measurement, Instrumentation and Automation*, vol. 336–338, pp. 332–335, Guilin, China, April 2013.
- [10] T. Abdelazim, W. Abdel-Hamid, N. El-Sheimy, and E. H. Shin, “Experimental results of an adaptive fuzzy network Kalman filtering integration for low cost navigation applications,” in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society: Fuzzy Sets in the Heart of the Canadian Rockies (NAFIPS '04)*, pp. 844–849, Banff, Canada, June 2004.
- [11] H. Yanling, Z. Yi, and S. Feng, “An anomaly recognition algorithm for DVL,” in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '11)*, pp. 2423–2427, IEEE, Beijing, China, August 2011.
- [12] Y. Zhang, Y. Ding, and N. Li, “A tightly integrated SINS/DVL navigation method for autonomous underwater vehicle,” in *Proceedings of the 5th International Conference on Computational and Information Sciences (ICCIS '13)*, pp. 1107–1110, Shiyang, China, June 2013.
- [13] K.-H. Tang, M.-M. Jiang, and J. Weng, “Design of SINS/Phased Array DVL integrated navigation system for underwater vehicle based on adaptive filtering,” *Journal of Chinese Inertial Technology*, vol. 21, no. 1, pp. 65–70, 2013 (Chinese).
- [14] Ø. Hegrehaes and O. Hallingstad, “Model-aided INS with sea current estimation for robust underwater navigation,” *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 316–337, 2011.
- [15] H. Miller and D. A. Hiltz, “Fault tolerant integrated inertial navigation/global positioning systems for next generation spacecraft,” in *Proceedings of the IEEE/AIAA 10th Digital Avionics Systems Conference*, pp. 207–212, Los Angeles, Calif, USA, October 1991.
- [16] T. Zhang, X.-S. Xu, Y. Li, and S.-P. Gong, “AUV Fault-tolerant technology based on inertial navigation and underwater acoustics assisted navigation system,” *Journal of Chinese Inertial Technology*, vol. 21, no. 4, pp. 512–516, 2013 (Chinese).
- [17] M. Ushaq and F. J. Cheng, “A fault tolerant integrated navigation scheme realized through online tuning of weighting factors for Federated Kalman Filter,” *Applied Mechanics and Materials*, vol. 446–447, pp. 1078–1085, 2014.
- [18] D. B. Fogel, “Introduction to simulated evolutionary optimization,” *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 3–14, 1994.
- [19] A. F. Tappenden and J. Miller, “A novel evolutionary approach for adaptive random testing,” *IEEE Transactions on Reliability*, vol. 58, no. 4, pp. 619–633, 2009.
- [20] M. L. M. Víctor, C. Vicent, L. Luis, and F. Anta Antonio, “A model of self-avoiding random walks for searching complex networks,” *Networks*, vol. 60, no. 2, pp. 71–85, 2012.
- [21] J. Du and R. Rada, “Dilemmas in knowledge-based evolutionary computation for financial investing,” *Intelligent Decision Technologies*, vol. 7, no. 2, pp. 123–136, 2013.
- [22] M. Mahsal Khan, A. Masood Ahmad, G. Muhammad Khan, and J. F. Miller, “Fast learning neural networks using Cartesian genetic programming,” *Neurocomputing*, vol. 121, pp. 274–289, 2013.
- [23] C. Xilin, “Neural network combined with evolutionary algorithm for Knowledge management in Electricity Supply Industry,” in *Proceedings of the IEEE International Symposium on Test and Measurement (ICTM '09)*, vol. 2, pp. 355–358, Hong Kong, 2009.
- [24] N. Mehdiyev and D. Enke, “Interest rate prediction: a neuro-hybrid approach with data preprocessing,” *International Journal of General Systems*, vol. 43, no. 5, pp. 535–550, 2014.
- [25] F. Zha, B.-Q. Hu, and J. Liu, “Prediction of gyro motor’s state based on grey theory and BP neural network,” *Journal of Chinese Inertial Technology*, vol. 18, no. 1, pp. 120–123, 2010 (Chinese).
- [26] G. Jing, W. Du, and Y. Guo, “Studies on prediction of separation percent in electro dialysis process via BP neural networks and improved BP algorithms,” *Desalination*, vol. 291, pp. 78–93, 2012.
- [27] Z. Haodong and L. Hongchan, “Optimized BP neural network model based on niche genetic algorithm,” in *Proceedings of the 2nd International Conference on Green Communications and Networks*, vol. 223 of *Lecture Notes in Electrical Engineering*, pp. 219–226, Springer, 2013.
- [28] L. Tian, Y. Luo, and Y. Wang, “Prediction model of TIG welding seam size based on BP neural network optimized by genetic algorithm,” *Journal of Shanghai Jiaotong University*, vol. 47, no. 11, pp. 1690–1696, 1701, 2013 (Chinese).
- [29] H. Yang, R. S. Fan, and D. H. Xu, “Power information system risk assessment method based on genetic algorithms and neural network,” *Applied Mechanics and Materials*, vol. 530–531, pp. 429–433, 2014.
- [30] H. Wei, *Theory and Method of the Neural Network Structure Design*, National Defence of Industry Press, Beijing, China, 2005.
- [31] P.-J. Li, X.-S. Xu, L.-H. Wang, and Y. Li, “Design of intelligent fault-tolerant to passive underwater integrated navigation system,” *Journal of Chinese Inertial Technology*, vol. 21, no. 2, pp. 221–225, 2013 (Chinese).



Hindawi

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

