Mechatronics 54 (2018) 155–165

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

In-move aligned SINS/GNSS system using recurrent wavelet neural network (RWNN)-based integration scheme☆

S. Rafatnia[a], H. Nourmohammadi[a,*], J. Keighobadi[a], M.A. Badamchizadeh[b]

[a] Department of Mechanical Engineering, University of Tabriz, Tabriz, Iran
[b] Department of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran

ARTICLE INFO

Keywords:
Data fusion
Recurrent wavelet neural network
In-move aligned SINS/GNSS
Low-cost navigation
MEMS-grade IMU

ABSTRACT

Advances in micro-electro mechanical system (MEMS) technology bring about revolutionary changes in autonomous vehicle navigation. As a new development, strap-down inertial navigation system (SINS) is effectively combined with global navigation satellite system (GNSS) to construct an integrated SINS/GNSS system. However, time-growing navigation error is the main challenge of using MEMS-grade inertial measurement unit (IMU) in the SINS/GNSS system. Failure of un-accounted inertial sensor error causes a rapid degradation in the overall performance of low-cost SINSs. This paper aims to enhance the long-term performance of low-cost MEMS-grade SINS/GNSS navigation system. A new integration scheme is presented for in-move aligned SINS/GNSS system. Unmodeled nonlinearities in the SINS dynamics as well as error uncertainties in the measurements of MEMS-grade IMU motivate using a robust data fusion algorithm for the proposed integration scheme. Considering these facts, a new recurrent wavelet neural network (RWNN)-based algorithm is designed for data fusion in the proposed in-move aligned SINS/GNSS system. Several vehicular field tests have been carried out to assess the long-term performance and accuracy of the proposed navigation algorithm.

1. Introduction

Alignment procedure in strap-down inertial navigation system (SINS) is performed in two steps known as initial alignment and in-move alignment. In the initial alignment, navigation parameters including position, velocity and orientation are initialized and corresponding direction cosine matrix (DCM) is constructed [1]. During the initial alignment, navigation block should be remained in a stationary mode prior to vehicle motion. Owing to poor initialization as well as the cumulative errors of inertial sensors (gyroscopes and accelerometers), initial alignment is not sufficient to guarantee a required level of navigation accuracy. Therefore, the SINS are often needed to be re-aligned during the vehicle motion especially, when micro-electro mechanical system (MEMS)-grade inertial measurement unit (IMU) is used in the SINS. The process of computing the true values of DCM matrix during the vehicle motion is called in-move alignment. In-move alignment is carried out based on position and velocity vector matching provided by an aiding system such as global navigation satellite system (GNSS) [2]. Complementary properties of SINS and GNSS systems motivate the design of an integrated and comprehensive in-move aligned SINS/GNSS navigation system. The ultimate aim is to obtain superior performance in the integrated system with respect to individual subsystems SINS and GNSS.

SINS navigation error arises from two main sources including structural errors in inertial sensors and computational errors due to numerical integration process. Deterministic errors in the inertial sensors can be removed during the calibration process. But, stochastic errors must be appropriately modelled to reduce adverse effects on position, velocity and orientation estimation [3]. As the main methodology for stochastic modelling of inertial sensors error, Gauss-Markov (GM) process is often applied to represent a large number of physical processes with relatively simple mathematical formulation [4]. Although first order GM process has been useful for modelling random errors of inertial sensors, but the performance of this method is related to choosing correlation time and variance of zero-mean white noise process. In addition, the autocorrelation of the random error in MEMS-grade sensors often seems to follow a higher order GM process [5].

SINS as a dead-reckoning navigation system is absolutely related to alignment of IMUs. Dealing with small misalignments, Kain and Cloutier presented a simple linear model for rapid transfer alignment for tactical application [6]. Scherzinger proposed psi-angle error model with large azimuth uncertainty [7]. Dimitriyev et al. developed an SINS error model with small tilt misalignment and large uncertainties [8]. He formulated the SINS alignment as a nonlinear filtering problem. Kong presented quaternion-based SINS error propagation model for in-motion

☆ This paper was recommended for publication by associate editor Dr. Luigi Del Re.

* Corresponding author.
E-mail address: Hnourmohammadi@tabrizu.ac.ir (H. Nourmohammadi).

https://doi.org/10.1016/j.mechatronics.2018.08.001
Received 15 November 2017; Received in revised form 22 June 2018; Accepted 13 August 2018
0957-4158/© 2018 Elsevier Ltd. All rights reserved.

alignment of low-cost SINS [9]. Hao et al. presented a particle filter for nonlinear in-move alignment of SINS with large initial attitude errors [10]. Ali and Ushaq developed a reliable in-move alignment scheme for low-cost SINS [11]. In this research work, robust Kalman filter structure has been used as the estimation algorithm in which measurement variables are provided by global positioning system (GPS) velocity data. Combining local observability concept with robust adaptive filtering, a new algorithm for rapid transfer alignment has been presented in [12]. Allotta et al. evaluated the performance of nonlinear Kalman filter including EKF and unscented Kalman filter (UKF) in the estimation process of inertial integrated navigation [13]. The performance of UKF in the in-move alignment of low-cost SINS has been also evaluated in [14,15]. Gan et al. presented a new algorithm for in-move alignment of IMUs by use of quaternion-based representation of misalignment errors [16]. Musavi and Keighobadi used Fuzzy neural network approach for approximation of IMU uncertainties [17]. Chang et al. developed odometer-aided in-move alignment algorithm for low-cost SINS [18]. They used a low-pass finite impulse response digital filter for disturbance attenuating in the odometer.

This research work aims to enhance the long-term performance and navigation accuracy of MEMS-grade SINS/GNSS systems. Integration mechanization and state estimation algorithm for data fusion between the SINS and the GNSS data are two substantial steps in developing an integrated SINS/GNSS system. In this research work, we concentrate on both the integration mechanization and state estimation algorithm. In the proposed integration mechanization, air-data calibration system in addition to vertical channel damping loop are integrated in an applied approach with in-move aligned SINS/GNSS system. Barometric altitude is computed in terms of pressure and temperature measured by corresponding sensors in the air-data calibration system. Then, the SINS dynamics is integrated with the vertical channel damping loop. Accordingly, the corrected values of altitude and vertical velocity of the host (under navigation) vehicle are estimated. As an advantage, the proposed approach provides appropriate measurements of vertical altitude and velocity in case of GNSS outage.

There are many researches that indicate the importance of providing accurate altitude in the long-term performance and reliability of SINS/GNSS systems. However, vertical channel damping loop integrated with calibrated air-data sensors are dropped in most previous researches. In the low-cost MEMS-grade SINSs, failure to compensate vertical channel instability errors results in exponentially divergence of computed altitude. Therefore, an integrated baro-inertial altimeter comprising of a damping loop and optimal state estimation mechanizations leads to compensated altitude error in vertical channel of navigation systems. In this paper, considering different environmental conditions in the form of standard and non-standard atmosphere models and using a MEMS altimeter, the barometric altitude is estimated and appropriately fed into the integrated SINS/GNSS.

State estimation algorithm for data fusion in the proposed integrated SINS/GNSS system is based on online recurrent wavelet neural network (RWNN) observer. As an advantage, the proposed RWNN algorithm is initialized without any knowledge of nonlinearities and bias uncertainties of the MEMS-grade inertial sensors. Moreover, unlike classical data fusion algorithm such as Kalman filter, the proposed algorithm is not limited to a zero-mean Gaussian white noise process with known covariance. Here, the RWNN learning rates are chosen in such a way that the stability of the observer is guaranteed based on Lyapunov's direct method.

The main contributions of the paper can be summarized as follows:

- Design of an applied integration mechanization for in-move aligned SINS/GNSS using baro-inertial altitude provided from air-data calibration system and vertical channel damping loop algorithm.
- Design of RWNN-based observer for data fusion in the proposed integrated SINS/GNSS system.
- Modeling inertial sensors' error by the proposed RWNN-based observer.
- Stability analysis of the proposed RWNN-based observer based on Lyapunov's direct method.
- Experimental evaluation of the proposed approach in several vehicular tests with different dynamical maneuverings.

2. Inertial navigation mechanization

As a major hardware simplification of old stable-platform navigation system, SINS overcomes the problems encountered with the old INS system, most importantly cost reduction results from the strap-down implementation, small size, light weight and low power consumption [19]. Inertial sensors play a key role in classifying the SINS as a high-performance SINS or low-cost SINS. Cumulative errors of MEMS-grade inertial sensors lead to time-increasing positioning error in low-cost inertial integrated navigation systems. The integration algorithm which is proposed for low-cost SINS should have an acceptable performance in estimation of inertial sensors errors.

2.1. SINS dynamic modelling

As shown in Fig. 1, the main coordinate frames in the inertial navigation systems are (1) Earth-centred Earth-fixed (e-frame), (2) inertial frame (*i*-frame), that has their origin at the centre of mass of the Earth and contrast to e-frame rotates in inertial space in order to remain fixed with respect to the surface of the earth, (3) body frame (*b*-frame) which has its origin at the centre of the inertial system, and (4) navigation frame (n-frame). Here, the n-frame is local level navigation frame with north, east and down (N-E-D) geodetics axes. The body xyz axes are coincided to N-E-D axes when the roll, pitch and heading angles are zero value.

The SINS latitude-longitude-altitude differential equations in the n-frame are expressed as follows [20]:

$$\dot{L} = \frac{v_N}{R_N + h}, \qquad \dot{l} = \frac{v_E}{(R_E + h)\cos L}, \qquad \dot{h} = -v_D \tag{1}$$

where, L, l and h are latitude, longitude, and altitude, respectively. The velocity components in the n-frame are characterized by v_N, v_E, and v_D. R_N and R_E stand for meridian and transverse radiuses of curvature. The velocity dynamics of the SINS in the n-frame are expressed as follows [20]:

$$\dot{v}_N = f_N - v_E\left(2\omega_e + \frac{v_E}{(R_E + h)\cos L}\right)\sin L + \frac{v_D v_N}{R_N + h} \tag{2}$$

$$\dot{v}_E = f_E - 2\omega_e\left(v_N \sin L + v_D \cos L\right) + \frac{v_E}{R_E + h}\left(v_D + v_N \sin L\right) \tag{3}$$

$$\dot{v}_D = f_D - v_E\left(2\omega_e + \frac{v_E}{(R_E + h)\cos L}\right)\cos L - \frac{v_N^2}{R_N + h} + g \tag{4}$$

where, ω_e is the magnitude of the Earth rate and g is the gravitational acceleration. Specific force vector in the n-frame is specified by $\boldsymbol{f}^n = [f_N \quad f_E \quad f_D]^T$ and is calculated from the accelerometer outputs in the b-frame, $\boldsymbol{f}^b$, as follows:

$$\boldsymbol{f}^n = \begin{bmatrix} f_N & f_E & f_D \end{bmatrix}^T = \boldsymbol{C}_b^n \boldsymbol{f}^b \tag{5}$$

According to z-y-x rotation sequence for Euler angles, the rotation/transformation matrix from the b-frame to the n-frame, $\mathbf{C}_b^n$ is expressed in terms of ψ (yaw angle about z-axis), θ (pitch angle about y-axis) and φ (roll angle about x-axis), as follows [21]:

$$\mathbf{C}_b^n = \begin{bmatrix} C\theta\, C\psi & -C\varphi\, S\psi + S\varphi\, S\theta\, C\psi & S\varphi\, S\psi + C\varphi\, S\theta\, C\psi \\ C\theta\, S\psi & C\varphi\, C\psi + S\varphi\, S\theta\, S\psi & -S\varphi\, C\psi + C\varphi\, S\theta\, S\psi \\ -S\theta & S\varphi\, C\theta & C\varphi\, C\theta \end{bmatrix} \tag{6}$$

where, C and S stand for cosine and sine functions, respectively. The rotation matrix is updated based on the following dynamics [22].

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{nb}^b \tag{7}$$

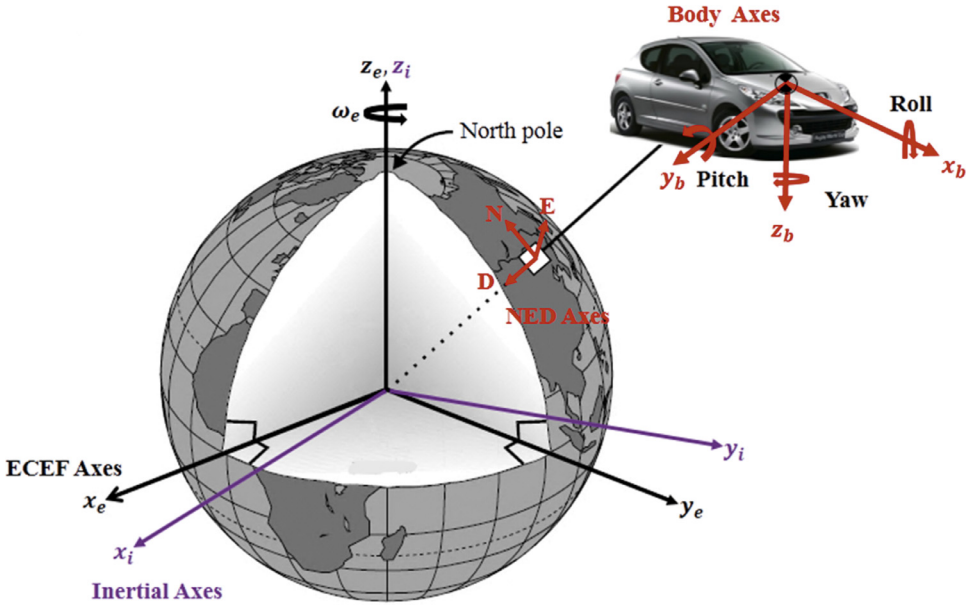

Fig. 1. Reference coordinate systems in the inertial navigation.

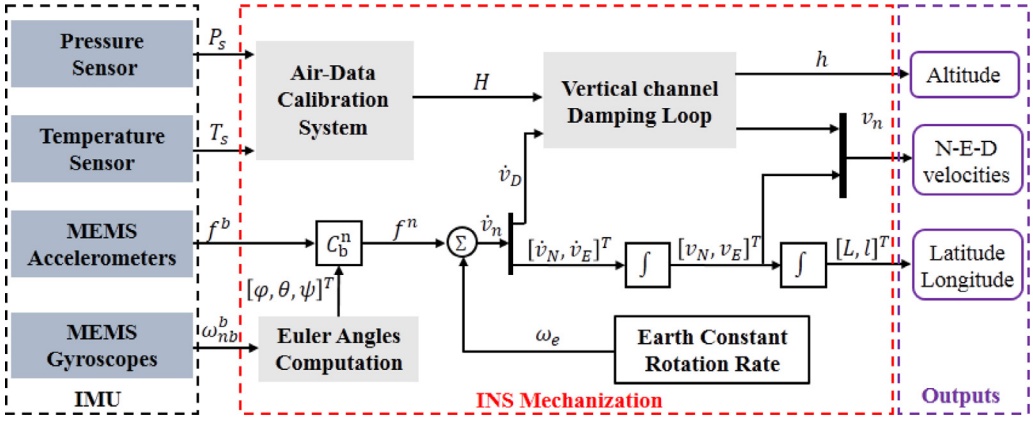

Fig. 2. Detailed block-diagram of the SINS with vertical channel damping loop.

where, $\mathbf{\Omega}^b_{nb}$ is the skew-symmetric matrix form of $\boldsymbol{\omega}^b_{nb}$ representing the rate of the b-frame relative to the n-frame. In low-cost SINS, $\mathbf{\Omega}^b_{nb}$ can be approximated into $\mathbf{\Omega}^b_{ib}$ as the gyroscope outputs.

2.2. Vertical channel damping loop

In the SINS, vertical channel instability leads to divergence in the altitude and vertical velocity estimation. Moreover, it can impress the other navigation states in the horizontal plane. Considering this fact, barometric altitude under a damping loop is proposed to compensate vertical channel errors and consequently prevent vertical channel instability. Block diagram of the proposed SINS with vertical damping loop is depicted in Fig. 2.

Barometric altitude is calculated in terms of pressure and temperature measured by corresponding sensors in air-data calibration system, as follows:

$$H_p = \frac{T_0}{L}\left[\left(\frac{P_s}{P_0}\right)^{-\frac{LR}{g}} - 1\right] + H_0 \tag{8}$$

where, the temperature and pressure at sea level are characterized by T_0 and P_0, respectively and in the standard atmosphere they are assumed to be 288.15 ($^\circ K$) and 101.325 (kPa). L, R, and g are the constant lapse rate, universal gas constant, and gravity constant, respectively. H_0 is taken 0 for sea level data, and P_s stands for barometer pressure. It must be noticed that (8) is only accurate for standard atmosphere condition. In non-standard atmosphere, the temperature and pressure assumptions cannot be guaranteed as real atmosphere. Applying scale factor, s and bias, b, barometric altitude compensation is accomplished as the following equation for non-standard atmosphere.

$$H = H_P + s\,(H_P - H_0) + b \tag{9}$$

where, H is the calibrated barometric altitude for non-standard atmosphere conditions. The scale factor and bias are defined as follows:

$$s = \frac{\Delta T}{T_0} \tag{10}$$

$$b = \frac{RT_0}{g}\left(\frac{\Delta P}{P_0}\right) \tag{11}$$

ΔT and ΔP are the deviation of true temperature and true pressure in the sea level from the standard condition, respectively. To calculate the true values of local sea level temperature and pressure around tests region, the following model is used.

$$\begin{aligned} T_t &= T_s + L\,H_P \\ P_t &= P_s + \rho g\,H_p \end{aligned} \tag{12}$$

where, T_t and P_t represent the true temperature and pressure in sea level; T_s and P_s are the values measured by sensors. The calibrated barometric altitude is included in the vertical channel dynamics. Accordingly, the following equations are derived as the vertical channel damping loop model.

$$\begin{aligned} h &= -v_D - K_1(h - H) \\ \dot{v}_D &= f_D - v_E\left(2\omega_e + \frac{v_E}{(R_E + h)\cos L}\right)\cos L - \frac{v_N^2}{R_N + h} \\ &\quad + g - K_2(h - H) - K_3\int (h - H)\,dt \end{aligned} \tag{13}$$

where, the suitable feedback gains of vertical damping loop are calculated as follows:

$$K_1 = 3\varepsilon \qquad K_2 = 4\varepsilon^2 + \frac{2g}{R} \qquad K_3 = 2\varepsilon^3 \tag{14}$$

and ε is commonly chosen, 0.01 [23].

2.3. SINS error dynamics

The standard form of SINS error dynamics is organized based on the dynamics of three error vectors including misalignment error, position error, and velocity error. The following equation is the state-space form of the SINS error model [24].

$$\delta\dot{\boldsymbol{x}} = \boldsymbol{W}\delta\boldsymbol{x} + \boldsymbol{G}\delta\boldsymbol{u} \tag{15}$$

where, the state vector, $\delta\boldsymbol{x}$ is constructed by nine states including three misalignment errors ($\delta\alpha$, $\delta\beta$, $\delta\gamma$), three velocity errors ($\delta v_N, \delta v_E, \delta v_D$), and three position errors ($\delta L, \delta l, \delta h$), as follows:

$$\delta\boldsymbol{x} = \begin{bmatrix}\delta\alpha & \delta\beta & \delta\gamma & \delta v_N & \delta v_E & \delta v_D & \delta L & \delta l & \delta h\end{bmatrix}^T \tag{16}$$

In (15), $\delta\boldsymbol{u}$ is the input noise vector containing gyroscope and accelerometer noises along x-y-z axes of the b-frame.

$$\delta\boldsymbol{u} = \begin{bmatrix}\delta\omega_x & \delta\omega_y & \delta\omega_z & \delta f_x & \delta f_y & \delta f_z\end{bmatrix}^T \tag{17}$$

The matrix $\boldsymbol{W}$ is the system error matrix defined as follows:

$$\boldsymbol{W} = \begin{bmatrix} W_{\Psi\Psi} & W_{\Psi v} & W_{\Psi r} \\ W_{v\Psi} & W_{vv} & W_{vr} \\ W_{r\Psi} & W_{rv} & W_{rr} \end{bmatrix} \tag{18}$$

Detailed description of the matrix $\boldsymbol{W}$ is presented in Appendix. The matrix $\boldsymbol{G}$ is the weight-matrix of the input noise vector and is expressed as follows [24]:

$$\boldsymbol{G} = \begin{pmatrix} -C^n_{b11} & -C^n_{b12} & -C^n_{b13} & 0 & 0 & 0 \\ -C^n_{b21} & -C^n_{b22} & -C^n_{b23} & 0 & 0 & 0 \\ -C^n_{b31} & -C^n_{b32} & -C^n_{b33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C^n_{b11} & C^n_{b12} & C^n_{b13} \\ 0 & 0 & 0 & C^n_{b21} & C^n_{b22} & C^n_{b23} \\ 0 & 0 & 0 & C^n_{b31} & C^n_{b32} & C^n_{b33} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{19}$$

Using (16) through (19) in (15), the SINS error state are calculated. Accordingly, the estimated position, velocity, and orientation are corrected based on the following equations.

$$\begin{aligned} L &= \hat{L} - \delta L \\ l &= \hat{l} - \delta l \\ h &= \hat{h} - \delta h \end{aligned} \tag{20}$$

$$\begin{aligned} v_N &= \hat{v}_N - \delta v_N \\ v_E &= \hat{v}_E - \delta v_E \\ v_D &= \hat{v}_D - \delta v_D \end{aligned} \tag{21}$$

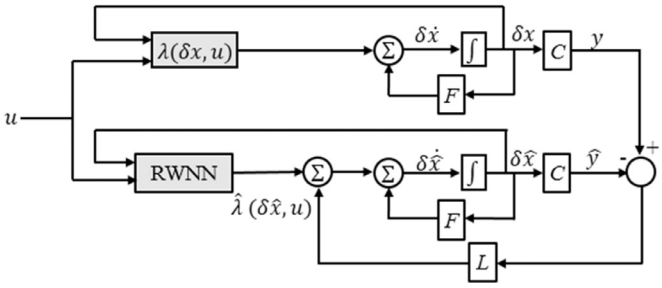

Fig. 3. Block diagram of RWNN approximation.

in which, the element with (ˆ) is the estimated quantity and that without (ˆ) is the corrected quantity. Furthermore, the DCM matrix is corrected by use of the misalignment angles based on Poisson equation.

$$\hat{\boldsymbol{C}}^n_b = [\boldsymbol{I} - \boldsymbol{\Psi}]\boldsymbol{C}^n_b \tag{22}$$

where, $\boldsymbol{I}$ stands for identity matrix and $\boldsymbol{\Psi}$ is the skew-symmetric matrix form of the misalignment vector, $[\delta\alpha\ \ \delta\beta\ \delta\gamma]^T$. Therefore, in-run correction of the DCM matrix and in-move alignment is carried out during the navigation process. However, due to large errors of the MEMS-grade inertial sensors as well as the cumulative computation errors, the SINS error states computed from (15) are erroneous. Hence, the SINS should be appropriately integrated with an auxiliary navigation system such as GNSS to achieve the required navigation accuracy and reliability.

3. RWNN-based observer design

State estimation algorithm for data fusion between the SINS and the GNSS data are one of the substantial steps in developing an integrated SINS/GNSS system. Standard Kalman filter is probably the most widely used algorithm for this purpose. However, it suffers from some drawbacks. For example, it is limited to a zero-mean Gaussian white noise process with known covariance. Here, a new integration scheme is designed for in-move aligned SINS/GNSS navigation system based on RWNN approach. As an advantage, the proposed RWNN algorithm is initialized without any knowledge of nonlinearities and bias uncertainties of the MEMS-grade inertial sensors. Moreover, it is not limited to a zero-mean Gaussian white noise process. In the following sections, the proposed RWNN-based observer is described in detail.

3.1. Observer design

The state-space model of the SINS error dynamics in (15) can be rewritten as:

$$\begin{aligned} \delta\dot{\boldsymbol{x}} &= \boldsymbol{W}\delta\boldsymbol{x} + \boldsymbol{G}\delta\boldsymbol{u} = \boldsymbol{F}\,\delta\boldsymbol{x} + \lambda(\delta\boldsymbol{x}, \boldsymbol{u}) \\ \boldsymbol{y} &= \boldsymbol{C}\delta\boldsymbol{x} \end{aligned} \tag{23}$$

The objective is to design an observer for minimizing the mismatch between the system output and the reference value. As shown in Fig. 3, the system dynamics with RWNN observer can be formulated as:

$$\begin{aligned} \delta\dot{\hat{\boldsymbol{x}}} &= W\delta\hat{\boldsymbol{x}} + \hat{\lambda}(\delta\hat{\boldsymbol{x}}, \boldsymbol{u}) + \boldsymbol{L}(\boldsymbol{y} - \hat{\boldsymbol{y}}) = \boldsymbol{F}\delta\hat{\boldsymbol{x}} + \boldsymbol{\tau}^T\boldsymbol{\Psi}_W(\delta\hat{\boldsymbol{x}}, \boldsymbol{u}, \hat{\xi}, \hat{\imath}, \hat{d}) + \boldsymbol{L}(\boldsymbol{y} - \hat{\boldsymbol{y}}) \\ \hat{\boldsymbol{y}} &= \boldsymbol{C}\delta\hat{\boldsymbol{x}} \end{aligned} \tag{24}$$

where, $\delta\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ are the estimated state vector and output vector, respectively. Using the estimated state vector, $\delta\hat{\boldsymbol{x}}$ and also the sensors output vector, $\boldsymbol{u}$ as the inputs to RWNN, the approximation of uncertainty vector, $\delta\boldsymbol{u}$ is achieved. In this approximation, $\boldsymbol{\tau}$ is output weight matrix; $\boldsymbol{\Psi}_W$ denotes wavelet vector; $\hat{\xi}$, $\hat{\imath}$, and $\hat{d}$ represent the input feedback weight, dilation and translation of wavelet respectively. The observer gain is characterized by $\boldsymbol{L}$ and is designed in such a way that the matrix $\tilde{\boldsymbol{A}} = \boldsymbol{F} - \boldsymbol{L}\,\boldsymbol{C}$ be a Hurwitz matrix. So, there exists $\boldsymbol{P} = \boldsymbol{P}^T > 0$ as the

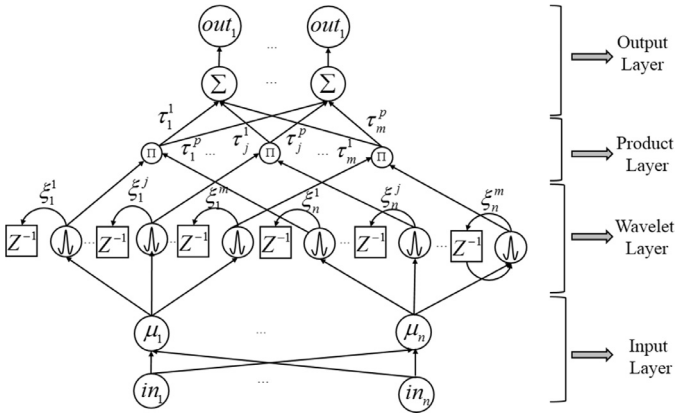

Fig. 4. The main structure of RWNN approximator.

solution of Lyapunov equation $\tilde{\boldsymbol{A}}^T\boldsymbol{P}+\boldsymbol{P}\tilde{\boldsymbol{A}}=-\boldsymbol{Q}$ with $\boldsymbol{Q}=\boldsymbol{Q}^T>0$. The RWNN approximated terms for simplicity is replaced by $\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u})$.

The state estimation error and also the output error are dynamically modelled by subtracting (24) from (23), as follow:

$$\dot{\delta\tilde{\boldsymbol{x}}}=(\boldsymbol{F}-\boldsymbol{LC})\delta\tilde{\boldsymbol{x}}+\left[\lambda(\delta\boldsymbol{x},\boldsymbol{u})-\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u})\right]=\tilde{\boldsymbol{A}}\delta\tilde{x}+\left[\lambda(\delta\tilde{\boldsymbol{x}},\boldsymbol{u})-\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u})\right]$$
$$\tilde{\boldsymbol{y}}=\boldsymbol{C}\delta\tilde{\boldsymbol{x}} \tag{25}$$

where, $\delta\tilde{\boldsymbol{x}}=\delta\boldsymbol{x}-\delta\hat{\boldsymbol{x}}$ and $\tilde{\boldsymbol{y}}=\boldsymbol{y}-\hat{\boldsymbol{y}}$ are the state and output errors. The term $(\lambda(\delta\boldsymbol{x},\boldsymbol{u})-\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u}))$ can be rewritten as follows:

$$\lambda(\delta\boldsymbol{x},\boldsymbol{u})-\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u})=\boldsymbol{\tau}^{*T}\boldsymbol{\Psi}_w(\delta\boldsymbol{x},\boldsymbol{u},\xi^*,t^*,d^*)+\boldsymbol{\varepsilon}-\boldsymbol{\tau}^T\boldsymbol{\Psi}_w(\delta\hat{\boldsymbol{x}},\boldsymbol{u},\hat{\xi},\hat{t},\hat{d}) \tag{26}$$

where, $\boldsymbol{\tau}^{*T}\boldsymbol{\Psi}(\delta\boldsymbol{x},\boldsymbol{u},\xi^*,t^*,d^*)+\boldsymbol{\varepsilon}$ is an ideal approximation set of $\lambda(\delta\boldsymbol{x},\boldsymbol{u})$, and $\boldsymbol{\varepsilon}$ shows the bounded approximation error ($\|\boldsymbol{\varepsilon}\|\le\varepsilon_m$ for positive real constant of ε_m).

3.2. RWNN approximation

As shown in Fig. 4, the RWNN approximator is designed as a four layer structure. First layer is input layer with normalized values; second layer is recurrent wavelet layer, third layer is product layer and fourth layer is output layer. This section presents the wavelet function structure and the corresponding learning algorithm. Here, the mother wavelet is used in Mexican-hat form, as follows:

$$\sigma(\boldsymbol{\mu}(s))=(1-||\boldsymbol{\mu}(s)||^2)e^{-||\mu(s)||^2/2} \tag{27}$$

where, $\boldsymbol{\mu}(s)\,\varepsilon\,[0,\,1]$ is the normalized input vector whereas $\|\boldsymbol{\mu}(s)\|^2=\boldsymbol{\mu}^T(s)\boldsymbol{\mu}(s)$. Detailed description of the proposed RWNN algorithm is presented in the following.

Layer 1 (Input layer):

In this layer, the input variables including sensors outputs and the last update of the state vector$in_i=(\boldsymbol{f}^b\quad\boldsymbol{\omega}_{nb}^b\quad\boldsymbol{H}\quad\tilde{\boldsymbol{x}})^T, i=1,2,\ldots,16$, are normalized to be in the interval of [0, 1]. So, in the block diagram of Fig. 4, there exists 16 nodes and each node corresponds to an input variable. The normalizing process is carried out as follows:

$$\mu_i=\frac{|in_i|}{\sum_{i=1}^{16}|in_i|},\quad i=1,2,\ldots,16 \tag{28}$$

where μ_i denotes the i-element of the normalized input vector to network.

Layer 2 (Wavelet layer):

In this layer, every incoming signals (μ_i) from layer 1 are crossed from three activation functions specified by j index. This activation functions are derived from mother wavelet of (27), as follows:

$$\sigma_{d,t}(\boldsymbol{\mu}(s))=2^{0.5d}\sigma(2^d\boldsymbol{\mu}(s)-t) \tag{29}$$

where, d and t are the wavelet dilation and translation, respectively. Then, the past information of the network is stored by use of the term $\sigma_{d,t}(Z^{-1}(s))$ in which the later memory term, $Z_{ij}^{-1}(s)$ is calculated from the previous memory term, $Z_{ij}^{-1}(s-1)$, as follows:

$$Z_{ij}^{-1}(s)=\mu_i(s)+\xi_i^j\sigma_{d_j,t_j}(Z_{ij}^{-1}(s-1)) \tag{30}$$

where, ξ_i^j stands for the recurrent wavelet layer feedback term. Eq. (30) can be rewritten as:

$$\begin{aligned}Z_{ij}^{-1}(s)&=\mu_i(s)+\xi_i^j2^{0.5d_{ij}}\sigma(2^{d_{ij}}Z_{ij}^{-1}-t_{ij})\\&=\mu_i(s)+\xi_i^j2^{0.5d_{ij}}(1-||2^{d_{ij}}Z_{ij}^{-1}(s-1)-t_{ij}||^2)\\&\quad e^{\frac{-||2^{d_{ij}}Z_{ij}^{-1}(s-1)-t_{ij}||^2}{2}}\end{aligned} \tag{31}$$

Layer 3 (Product layer):

Each neuron of this layer specified by Π in Fig. 4, is multiplying of incoming signals from layer 2 as follow:

$$\kappa_j=\Pi\sigma_{d_{ij},t_{ij}}(Z_{ij}^{-1}(s)) \tag{32}$$

Layer 4 (Output Layer):

In this layer, the final output of RWNN is calculated based on the following equation.

$$\hat{\lambda}(\delta\hat{\boldsymbol{x}},\boldsymbol{u})=out_l=\sum_{j=1}^{3}\boldsymbol{\tau}_j^l\kappa_j\quad l=1,2,\ldots,6 \tag{33}$$

where, out_l is the approximated uncertainty vector and $\boldsymbol{\tau}_j^l$ denotes the connection weight of the output layer.

3.3. Learning algorithm of the RWNN

In this section an online learning algorithm is presents for the proposed RWNN-based observer. The algorithm consists of parameter and structure learning. The learning algorithm of dilation and translation of mother wavelet (d and t), recurrent wavelet layer feedback term (ξ_i^j) and connection weight of output layer ($\boldsymbol{\tau}_j^l$) is based on back propagation method. In this method, the objective is to minimizing the cost function based on the gradient descent method. The mismatch between the reference and estimated values are formulated in accordance with the following cost function.

$$J_O(s)=\frac{1}{2}\tilde{\boldsymbol{y}}^T\tilde{\boldsymbol{y}}=0.5\sum_{l=1}^{6}(\boldsymbol{y}_l(s)-\hat{\boldsymbol{y}}_l(s))^2 \tag{34}$$

As a quadratic function, the cost function of (34) has a single and unique minimum solution. The updating process of wavelet parameters is described as follows.

3.3.1. Update law in layer 2

The update law of the wavelet translation (t_{ij}) is constructed based on the following procedure.

$$\begin{aligned}\Delta t_{ij}&=-\eta_t\frac{\partial J_o}{\partial t_{ij}}=-\eta_t\frac{\partial J_o}{\partial y_l}\frac{\partial\hat{y}_l}{\partial\hat{\lambda}_l}\frac{\partial\hat{\lambda}_l}{\partial\kappa_j}\frac{\partial\kappa_j}{\partial t_{ij}}\\&=-\eta_t\sum_{l=1}^{6}[y_l(s)-\hat{y}_l(s)]\left\{\sum_{j=1}^{3}\tau_j^l\frac{\partial\kappa_j}{\partial t_{ij}}\right\}\end{aligned} \tag{35}$$

where, $\frac{\partial\kappa_j}{\partial t_{ij}}$ can be rewritten as follows:

$$\begin{aligned}\frac{\partial\kappa_j}{\partial t_{ij}}&=\sigma(Z_{1j}^{-1})\sigma(Z_{2j}^{-1})\ldots\frac{\partial\sigma(Z_{ij}^{-1})}{\partial t_{ij}}=\frac{\kappa_j}{\sigma(Z_{ij}^{-1})}\frac{\partial\sigma(Z_{ij}^{-1})}{\partial t_{ij}}\\&=\frac{\kappa_j}{2^{d_{ij}/2}(1-T_{ij}^2)e^{-T_{ij}^2/2}}\{-2^{d_{ij}/2}T_{ij}(3-T_{ij}^2)e^{-T_{ij}^2/2}\}\\&=\kappa_j\frac{T_{ij}}{(1-T_{ij}^2)}(3-T_{ij}^2)\end{aligned} \tag{36}$$

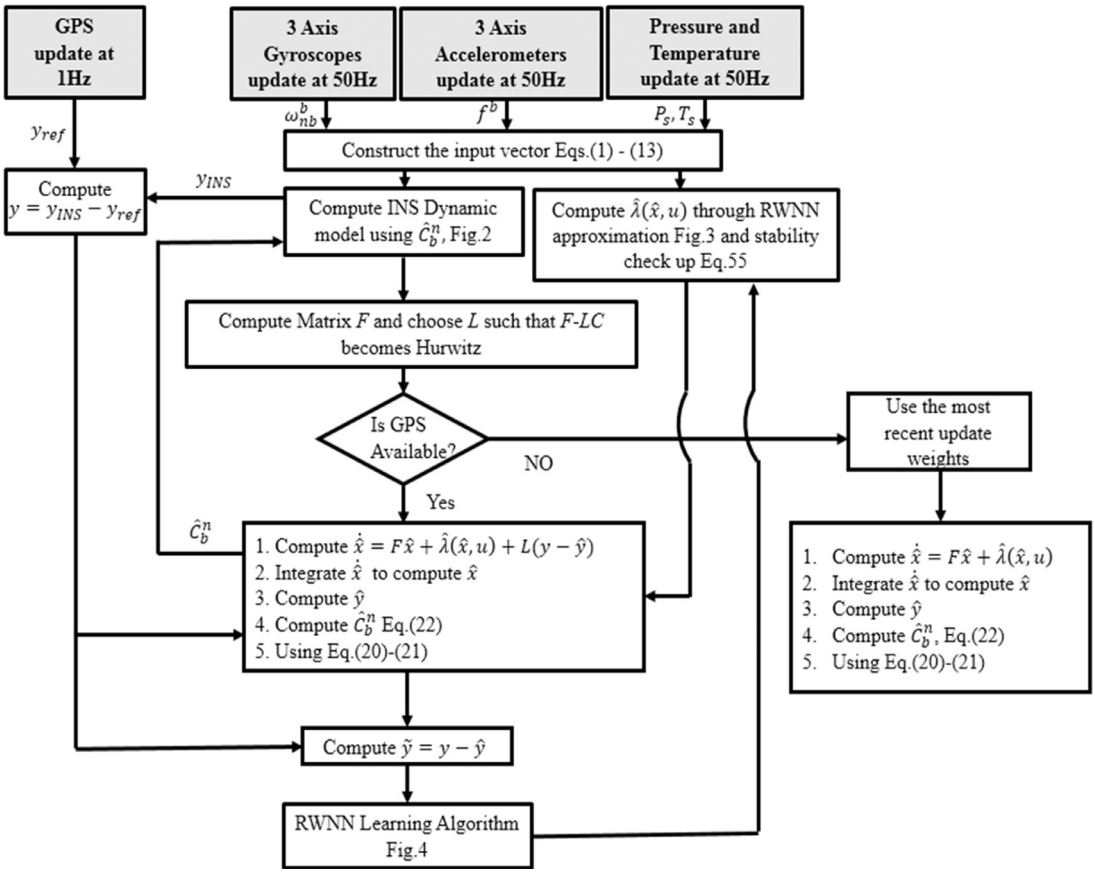

Fig. 5. RWNN implementation to in-move aligned SINS/GNSS.

where, $T_{ij} = 2^{d_{ij}} z_{ij}^{-1}(s) - t_{ij}$. Finally, the update law of t_{ij} is defined as follows:

$$t_{ij}(s+1) = t_{ij}(s) + \Delta t_{ij} \tag{37}$$

The update law of the wavelet dilation (d_{ij}) is constructed based on the following procedure.

$$\begin{aligned}\Delta d_{ij} &= -\eta_t \frac{\partial J_o}{\partial d_{ij}} = -\eta_d \frac{\partial J_o}{\partial y_l}\frac{\partial \hat{y}_l}{\partial \hat{\lambda}_l}\frac{\partial \hat{\lambda}_l}{\partial \kappa_j}\frac{\partial \kappa_j}{\partial d_{ij}} \\ &= -\eta_d \sum_{l=1}^{6}[y_l(s) - \hat{y}_l(s)]\left\{\sum_{j=1}^{3}\tau_j^l \frac{\partial \kappa_j}{\partial d_{ij}}\right\}\end{aligned} \tag{38}$$

where,

$$\frac{\partial \kappa_j}{\partial d_{ij}} = \kappa_j \left\{\frac{\ln 2}{2} - 2^{d_{ij}}(\ln 2) Z_{ij}^{-1} T_{ij}\left(1 + \frac{2}{1 - T_{ij}^2}\right)\right\} \tag{39}$$

and the update law of d_{ij} is defined as follows:

$$d_{ij}(s+1) = d_{ij}(s) + \Delta d_{ij} \tag{40}$$

The update law of the feedback term (ξ_i^j) is constructed based on the following procedure.

$$\begin{aligned}\Delta \xi_i^j &= -\eta_\xi \frac{\partial J_o}{\partial \xi_i^j} = -\eta_\xi \frac{\partial J_o}{\partial y_l}\frac{\partial \hat{y}_l}{\partial \hat{\lambda}_l}\frac{\partial \hat{\lambda}_l}{\partial \kappa_j}\frac{\partial \kappa_j}{\partial \xi_i^j} \\ &= -\eta_\xi \sum_{l=1}^{6}[y_l(s) - \hat{y}_l(s)]\left\{\sum_{j=1}^{3}\tau_j^l \frac{\partial \kappa_j}{\partial \xi_i^j}\right\}\end{aligned} \tag{41}$$

where,

$$\frac{\partial \kappa_j}{\partial \xi_i^j} = \kappa_j \left\{\frac{T_{ij}}{(1 - T_{ij}^2)}\sigma_{d_j,t_j}(Z_{ij}{}^{-1}(s-1))2^{d_{ij}}(T_{ij}^2 - 3)\right\} \tag{42}$$

and the update law of ξ_i^j is defined as follows:

$$\xi_i^j(s+1) = \xi_i^j(s) + \Delta \xi_i^j \tag{43}$$

3.3.2 Update law in layer 3

The update law of outputs connection weight is constructed based on the following procedure.

$$\Delta \tau_j^l = -\eta_\tau \frac{\partial J_o}{\partial \tau_j^l} = -\eta_\tau \frac{\partial J_o}{\partial y_l}\frac{\partial \hat{y}_l}{\partial \hat{\lambda}_l}\frac{\partial \hat{\lambda}_l}{\partial \omega_j^l} = -\eta_\tau \sum_{l=1}^{6}[y_l(s) - \hat{y}_l(s)]\kappa_j \tag{44}$$

and then,

$$\tau_j^l(s+1) = \tau_j^l(s) + \Delta \tau_j^l \tag{45}$$

3.4. Stability analysis

The stability analysis of the proposed RWNN observer is conducted via Lyapunov direct method. The following Lyapunov candidate is considered.

$$V(s) = J_O(s) = \frac{1}{2}\tilde{y}^T\tilde{y} = 0.5\sum_{l=1}^{6}(y_l(s) - \hat{y}_l(s))^2 = \frac{1}{2}e^2(s) \tag{46}$$

The error difference of the above Lyapunov candidate can be written as follows:

$$\Delta V = V(s+1) - V(s) = \frac{1}{2}[e^2(s+1) - e^2(s)] \tag{47}$$

where, $e(s+1)$ is the next iteration of the estimation error defined as follows:

$$e(s+1) = e(s) + \Delta e(s) \tag{48}$$

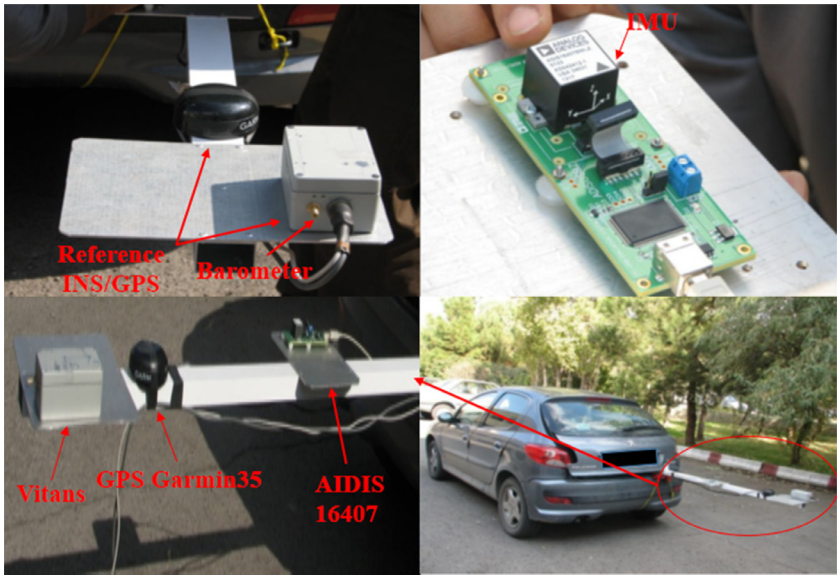

Fig. 6. Vehicular navigation test.

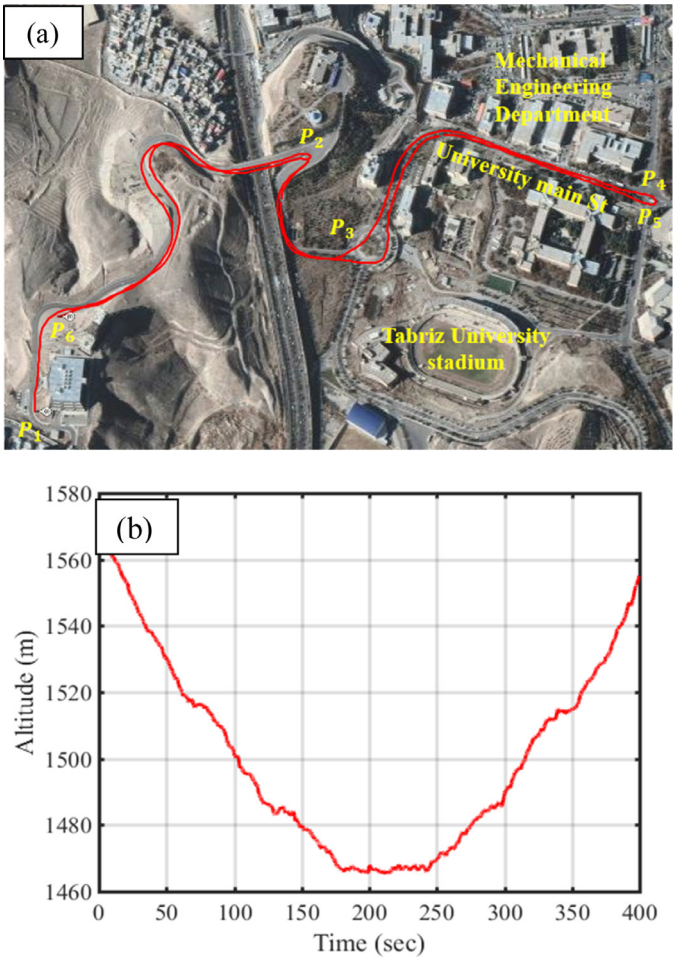

Fig. 7. (a) Reference latitude-longitude trajectory of test #1, (b) Reference altitude trajectory of test #1.

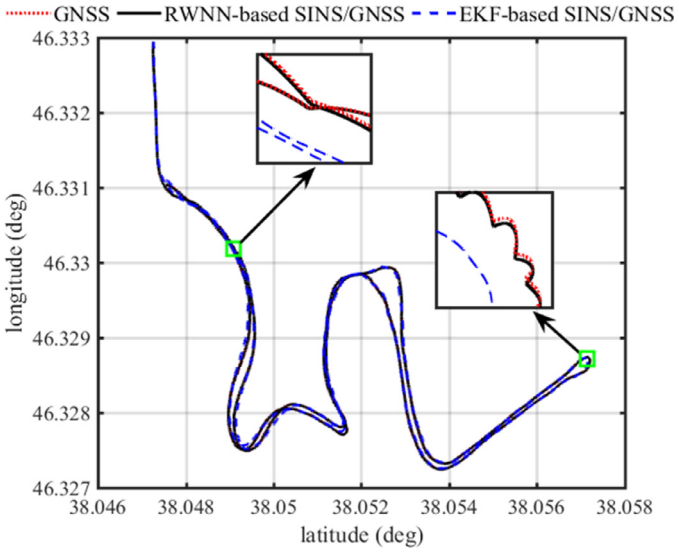

Fig. 8. Estimated trajectory through both the RWNN and EKF-based integrated navigation algorithms compared to the reference GNSS data during test#1.

Using (48) in (46) results in:

$$\Delta V = \frac{1}{2}[\Delta e^2(s) + 2e(s)\Delta e(s)] = \frac{1}{2}\Delta e(s)[\Delta e(s) + 2e(s)] \tag{49}$$

where, $\Delta e(s)$ is calculated as follows:

$$\Delta e(s) = \frac{\partial e}{\partial w}\Delta w \tag{50}$$

By defining $w = [t_{ij}, d_{ij}, \xi_i^j, \tau_j^l]$, Δw is expressed as:

$$\Delta w = -\eta_w \frac{\partial J_o}{\partial w} = -\eta_w e \frac{\partial y}{\partial w} \tag{51}$$

Therefore, (49) can be rewritten as follows:

$$\begin{aligned}\Delta V &= -\frac{1}{2}\frac{\partial e}{\partial w}\left(\eta_w e \frac{\partial y}{\partial w}\right)\left[-\frac{\partial e}{\partial w}\left(\eta_w e \frac{\partial y}{\partial w}\right) + 2e\right] \\ &= e^2\left[\frac{1}{2}\left(\frac{\partial e}{\partial w}\eta_w \frac{\partial y}{\partial w}\right)^2 - \frac{\partial e}{\partial w}\eta_w \frac{\partial y}{\partial w}\right]\end{aligned} \tag{52}$$

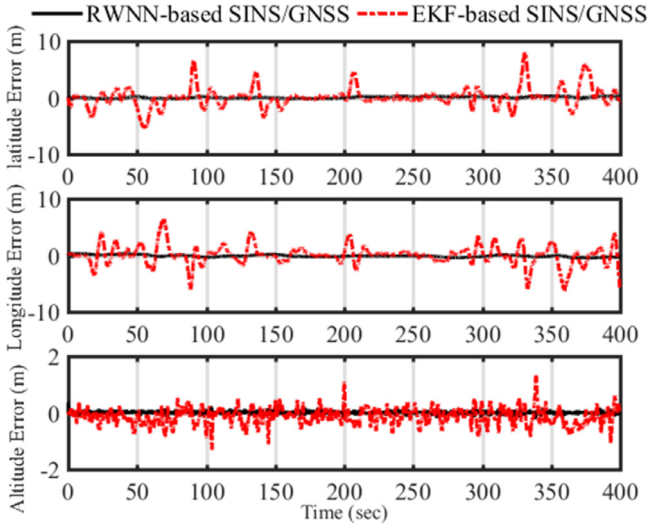

Fig. 9. Estimation error of the position components through the proposed RWNN-based algorithm compared to EKF-based algorithm, during test #1.

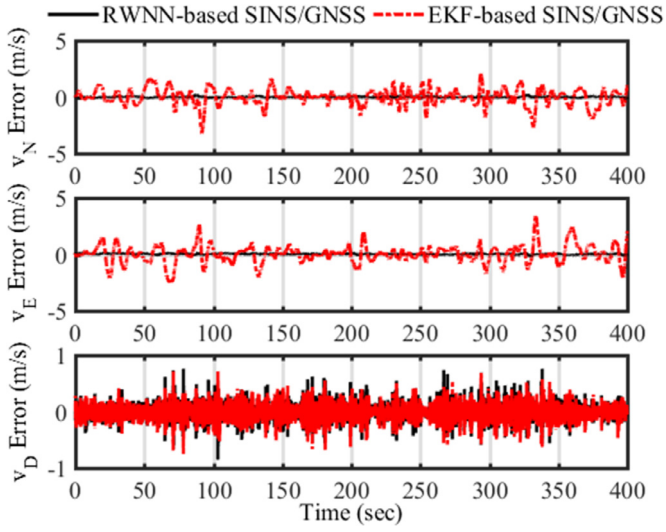

Fig. 10. Estimation error of the velocity components through the proposed RWNN-based algorithm compared to EKF-based algorithm, during test #1.

Using the fact that $\frac{\partial e}{\partial w} = \frac{\partial y}{\partial w}$, (52) is reformulated as:

$$\Delta V = e^2\left\{\frac{1}{2}\left[\left(\frac{\partial y}{\partial w}\right)^4 \eta_w^2\right] - \left(\frac{\partial y}{\partial w}\right)^2 \eta_w\right\}$$
$$= e^2\eta_w\left(\frac{\partial y}{\partial w}\right)^4\left[\frac{1}{2}\eta_w - \frac{1}{\left(\frac{\partial y}{\partial w}\right)^2}\right] \quad (53)$$

Knowing that $\eta_w > 0$, ΔV is negative-definite subjected to the following inequality.

$$0 < \eta_w < \frac{2}{\left(\frac{\partial y}{\partial w}\right)^2} \quad (54)$$

$V(s)$ can be a Lyapunov function and asymptotical stability is guaranteed. Therefore, by defining Y as:

$$\mathrm{Y} = \max\left(\left|\frac{\partial y}{\partial d}\right|, \left|\frac{\partial y}{\partial t}\right|, \left|\frac{\partial y}{\partial \xi}\right|, \left|\frac{\partial y}{\partial \omega}\right|\right) \quad (55)$$

asymptotical stability will be guaranteed by $0 < \eta_w < \frac{2}{\mathrm{Y}^2}$.

Table 1
Main Specifications of inertial sensors in ADIS-16407.

Parameter	Gyroscope	Accelerometer
Misalignment (axis-to-frame)	0.05°	0.2°
Misalignment (axis-to-axis)	0.5°	0.5°
Initial bias error (1σ)	3°/s	50 mg
In-run bias stability (1σ)	0.007°/s	0.2 mg
Random walk (1σ)	$1.9°/\sqrt{\text{hr}}$	$0.2\,\text{m/s}/\sqrt{\text{hr}}$
Output noise (no filtering)	0.8°/s rms	9 mg rms
Dynamic range	±300°/s	±18 g

3.5. Implementation on In-move aligned SINS/GPS

The proposed RWNN-observer is used for the integration scheme of in-move aligned SINS/GNSS. The system consists of 3-axis gyroscopes, 3-axis accelerometers, barometer, and thermometer with 50 Hz updating rate. The reference position and velocity data of the under-navigation vehicle is generated by a GPS receiver with 1 Hz updating rate. The implementation flowchart of the proposed RWNN is presented in Fig. 5.

As shown in Fig. 5, the IMU and environment sensors data are used as the input of the RWNN approximator. The measurement data are provided from GNSS position and velocity. The GNSS updating rate is very small compared to IMUs, barometer and thermometer updating rate. So, when GNSS is exist, the RWNN learning algorithm is done and in 1 s GNSS gap the system used the last update weight to approximation of IMUs uncertainties.

4. Experimental results and discussion

In this section, the proposed RWNN-based in-move aligned SINS/GNSS is experimentally verified through vehicular field tests performed by experienced colleagues in University of Tabriz. Test equipment is represented in Fig. 6. Inertial measurements are provided by ADIS-16407 as a MEMS-grade IMU. Temperature and raw pressure data are supplied by corresponding sensors in ADIS16407 system. Garmin-35 GPS receiver has been used to produce the measurement/output vector for the proposed integration scheme. Moreover, highly accurate Vitans navigation system is used to provide reference values for evaluation. The main statistical specifications of inertial sensors in ADIS-16407 IMU are given in Table 1.

Two vehicular field tests have been executed to assess the performance of the proposed in-move aligned SINS/GNSS system. Fig. 7 represents the reference geographical latitude- longitude trajectory and also the altitude trajectory during test #1.

Test #1 has been executed approximately for 400 seconds along a trajectory with wide range dynamics manoeuvring. According to Fig. 7(a), test #1 is started in P_1 and continues up to P_6 with several circulating manoeuvres for example in P_2 and P_4. The reference altitude in Fig. 7(b) shows that the vehicle has an ascending direction in test #1.

To legitimize the assessment of the proposed RWNN-based in-move aligned SINS/GNSS algorithm, the results are compared with those of a conventional SINS/GNSS system that uses standard EKF as the state estimation algorithm. Fig. 8 shows the estimated latitude-longitude position of the vehicle through both the RWNN and EKF-based integrated navigation algorithms, during test #1.

The estimation errors of the position and velocity components with respect to the reference GNSS data are shown in Figs. 9 and 10, respectively.

For better evaluation of the proposed RWNN-based in-move aligned SINS/GNSS algorithm, the mean value and the standard deviation of the position and velocity estimation errors corresponding to both the proposed and the conventional EKF-based SINS/GNSS algorithms are accumulated in Table 2. The quantitative results of Table 2 clearly shows that the proposed RWNN-based in-move aligned SINS/GNSS system yields to

Table 2
Mean value and standard deviation of the estimation error in test #1.

Navigation parameter	In-move aligned SINS/GNSS with RWNN-based integration		Conventional SINS/GNSS with EKF-based integration	
	Mean value of estimation error	Standard deviation of estimation error ($\pm 1\sigma$)	Mean value of estimation error	Standard deviation of estimation error ($\pm 1\sigma$)
Latitude error (m)	0.0303	0.0508	0.1280	0.2630
Longitude error (m)	−0.0853	0.0475	0.1559	0.3560
Altitude error (m)	0.00092	0.0418	−0.1336	0.2934
V-north error (m/s)	−0.0045	0.0542	−0.0470	1.7211
V-east error (m/s)	−0.0051	0.0552	0.0206	1.7542
V-down error (m/s)	0.00024	0.0952	−0.0094	0.1235

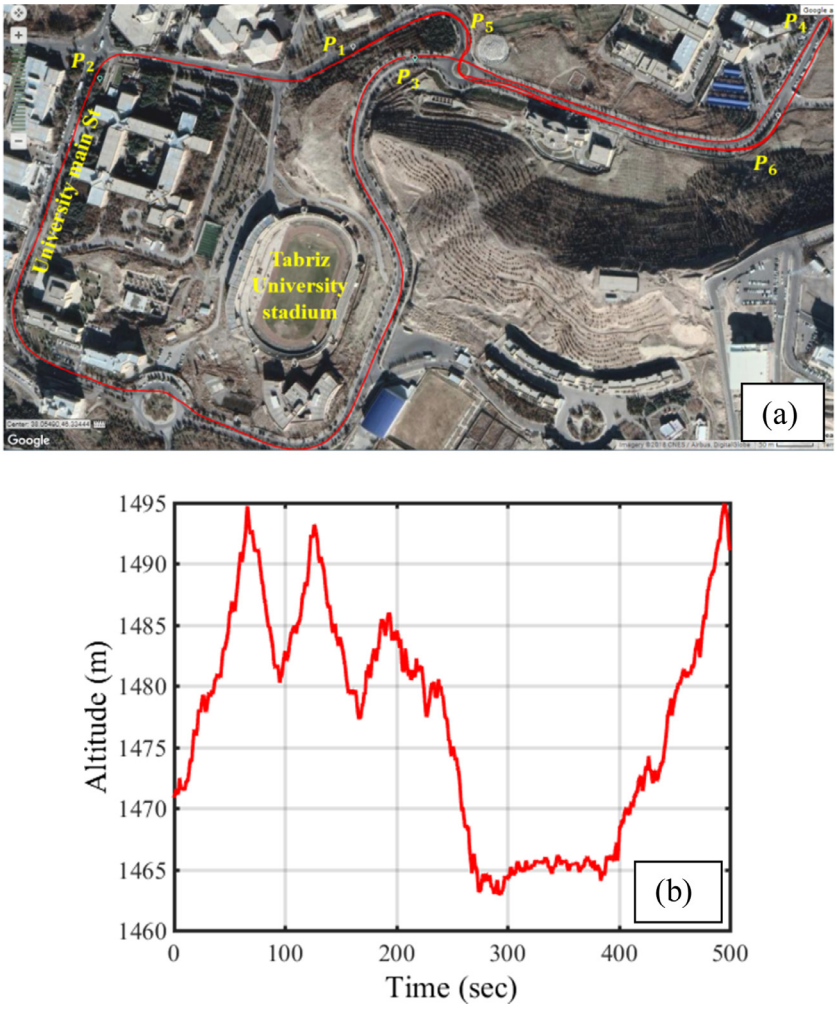

Fig. 11. (a) Reference latitude-longitude trajectory of test #2, (b) Reference altitude trajectory of test #2.

superior position and velocity estimation in comparison with the conventional EKF-based SINS/GNSS system.

To legitimize the performance and accuracy assessment of the proposed algorithm, another vehicular test has been executed in university of Tabriz. Test #2 takes long 500 seconds, approximately. Fig. 11 shows the reference geographical latitude-longitude trajectory and reference altitude trajectory of the vehicle during test #2. As shown in Fig. 11(b), the vehicle undergoes more altitude manoeuvring in test#2 compared to test #1. In test #2, the vehicle starts its move in P_1 and continues up to P_6. The distance between P_5 and P_6 is travelled twice by vehicle.

Fig. 12 shows the estimated latitude-longitude position of the vehicle through both the RWNN and EKF-based integrated navigation algorithms, during test #2. The estimation errors of the position and velocity components with respect to the reference GNSS data are shown in Figs. 13 and 14, respectively. The mean value and the standard deviation of position and velocity estimation errors corresponding to both the proposed and the conventional EKF-based SINS/GNSS algorithms in test #2 are accumulated in Table 3.

Statistical analysis of the estimation errors in Table 3 shows that using RWNN-based integration scheme in In-move aligned SINS/GNSS system significantly decreases the mean value and standard deviation of position and velocity estimation error compared to the conventional SINS/GNSS system with EKF-based integration scheme.

Table 3
Mean value and standard deviation of the estimation error in test #2.

Navigation parameter	In-move aligned SINS/GNSS with RWNN-based integration		Conventional SINS/GNSS with EKF-based integration	
	Mean value of estimation error	Standard deviation of estimation error ($\pm 1\sigma$)	Mean value of estimation error	Standard deviation of estimation error ($\pm 1\sigma$)
Latitude error (m)	0.0260	0.1115	0.0406	1.2420
Longitude error (m)	−0.0220	0.2259	−0.056	1.1285
Altitude error (m)	−0.00058	0.0398	−0.0885	0.2463
V-north error (m/s)	−0.0033	0.0377	−0.0101	0.5176
V-east error (m/s)	0.0240	0.0370	0.0027	0.5449
V-down error (m/s)	−0.00017	0.1161	−0.0060	0.1190

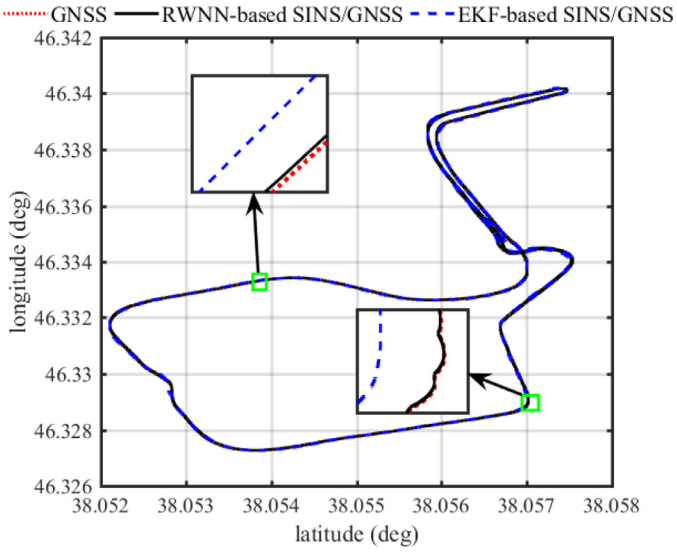

Fig. 12. Estimated trajectory through both the RWNN and EKF-based integrated navigation algorithms compared to the reference GNSS data during test#2.

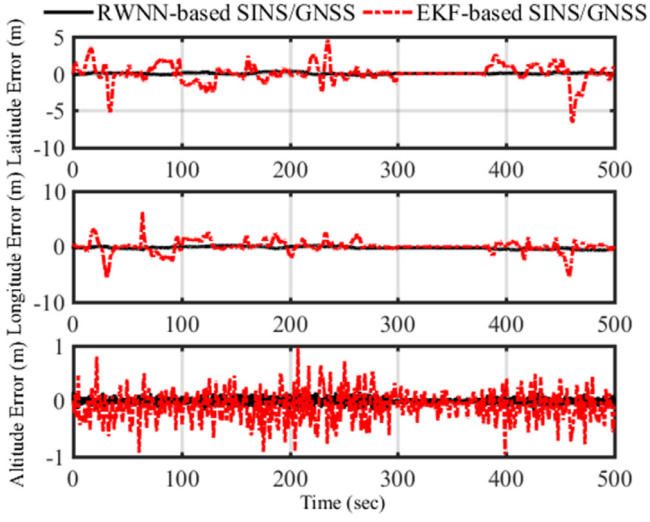

Fig. 13. Estimation error of the position components through the proposed RWNN-based algorithm compared to EKF-based algorithm, during test #2.

5. Conclusion

High performance inertial navigation systems could not be used in common civil applications owing to high costs as well as the regulation by governments. On the other hand, cumulative errors of MEMS-grade inertial sensors lead to time-increasing positioning error in low-cost inertial integrated navigation systems. Motivated by improving navigation accuracy, reliability and long-term performance of low-cost navigation systems, this research presented a new integration scheme for in-move aligned SINS/GNSS integrated navigation system. In the proposed algorithm, the integration between the SINS and GNSS has been carried out by use of RWNN-based observer. The proposed algorithm has been assessed in several vehicular field tests with wide range dynamical manoeuvring containing significant changes in acceleration, angular velocity, altitude, and heading angle. The proposed RWNN-based in-move aligned SINS/GNSS has been also evaluated in comparison with a conventional SINS/GNSS that uses EKF as the data fusion algorithm. It can be inferred from the statistical analysis of the experimental results that the proposed algorithm significantly enhances the overall navigation accuracy of low-cost SINS/GNSS system. Considering the theoretical and practical superiorities compared to EKF-based algorithm, the proposed RWNN-based data fusion is more suitable for implementation in low-cost SINS/GNSS integrated navigation systems.

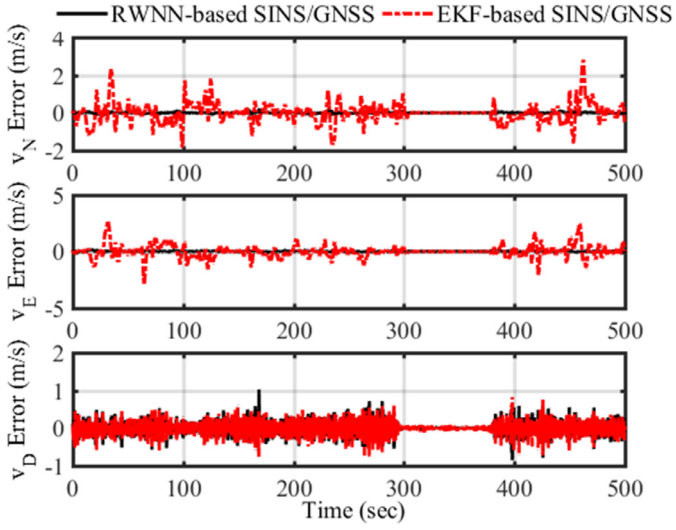

Fig. 14. Estimation error of the velocity components through the proposed RWNN-based algorithm compared to EKF-based algorithm, during test #2.

Appendix. Each elements of matrix *w* in the error dynamics of SINS are constructed as follows

$$W_{\Psi\Psi} = \begin{bmatrix} 0 & -\left(\omega_e \sin L + \frac{v_E}{R_0}\tan L\right) & \frac{v_N}{R_0} \\ \left(\omega_e \sin L + \frac{v_E}{R_0}\tan L\right) & 0 & \omega_e \cos L + \frac{v_E}{R_0} \\ -\frac{v_N}{R_0} & -\omega_e \cos L - \frac{v_E}{R_0} & 0 \end{bmatrix} \quad (A.1)$$

$$W_{\Psi v} = \begin{bmatrix} 0 & \frac{1}{R_0} & 0 \\ -\frac{1}{R_0} & 0 & 0 \\ 0 & -\frac{\tan L}{R_0} & 0 \end{bmatrix} \quad (A.2)$$

$$W_{\Psi r} = \begin{bmatrix} -\omega_e \sin L & 0 & -\frac{\upsilon_E}{R_0^2} \\ 0 & 0 & \frac{\upsilon_N}{R_0^2} \\ -\omega_e \cos L - \frac{\upsilon_E}{R_0 \cos^2 L} & 0 & \frac{\upsilon_E \tan L}{R_0^2} \end{bmatrix} \tag{A.3}$$

$$W_{\upsilon\Psi} = \begin{bmatrix} 0 & -f_D & f_E \\ f_D & 0 & -f_N \\ -f_E & f_N & 0 \end{bmatrix} \tag{A.4}$$

$$W_{\upsilon\upsilon} = \begin{bmatrix} \frac{\upsilon_D}{R_0} & -2\left(\omega_e \sin L + \frac{\upsilon_E}{R_0} \tan L\right) & \frac{\upsilon_E}{R_0} \\ 2\omega_e \sin L + \frac{\upsilon_E}{R_0} \tan L & \frac{1}{R_0}\left(\upsilon_N \tan L + \upsilon_D\right) & 2\omega_e \cos L + \frac{\upsilon_E}{R_0} \\ -\frac{2\upsilon_N}{R_0} & -2\left(\omega_e \cos L + \frac{\upsilon_E}{R_0}\right) & 0 \end{bmatrix} \tag{A.5}$$

$$W_{\upsilon r} = \begin{bmatrix} -\upsilon_E\left(2\omega_e \cos L + \frac{\upsilon_E}{R_0 \cos^2 L}\right) & 0 & \frac{1}{R_0^2}\left(\upsilon_E^2 \tan L - \upsilon_N \upsilon_D\right) \\ 2\omega_e(\upsilon_N \cos L - \upsilon_D \sin L) + \frac{\upsilon_N \upsilon_E}{R_0 \cos^2 L} & 0 & -\frac{\upsilon_E}{R_0^2}(\upsilon_N \tan L + \upsilon_D) \\ 2\omega_e \upsilon_E \sin L & 0 & \frac{1}{R_0^2}(\upsilon_N^2 + \upsilon_E^2) \end{bmatrix} \tag{A.6}$$

$$W_{r\psi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad W_{r\upsilon} = \begin{bmatrix} \frac{1}{R_0} & 0 & 0 \\ 0 & \frac{1}{R_0 \cos L} & 0 \\ 0 & 0 & -1 \end{bmatrix},$$

$$W_{rr} = \begin{bmatrix} 0 & 0 & -\frac{\upsilon_E}{R_0^2} \\ \frac{\upsilon_E \tan L}{R_0 \cos L} & 0 & -\frac{\upsilon_E}{R_0^2 \cos L} \\ 0 & 0 & 0 \end{bmatrix} \tag{A.7}$$

where, R_0 is calculated as $R_0 = \sqrt{R_N R_E}$.

References

[1] Cao S, Guo L. Multi-objective robust initial alignment algorithm for Inertial Navigation System with multiple disturbances. Aerospace Sci Technol 2012;21(September(1)):1–6.
[2] Nourmohammadi H, Keighobadi J. Integration scheme for SINS/GPS system based on vertical channel decomposition and in-motion alignment. AUT J Model Simul 2018;50(June(1)):11–20.
[3] Noureldin A, Karamat TB, Eberts MD, et al. Performance enhancement of MEMS-based INS/GPS integration for low-cost navigation applications. IEEE Trans Veh Technol 2009;58(March(3)):1077–96.
[4] Cohen O, Edan Y. A sensor fusion framework for online sensor and algorithm selection. Robot Auton Syst. 2008;56(September(9)):762–76.
[5] Nourmohammadi H, Keighobadi J. Design and experimental evaluation of indirect centralized and direct decentralized integration scheme for low-cost INS/GNSS system. GPS Sol. 2018;22(April(65)):1–18.
[6] Kain JE, Cloutier JR. Rapid transfer alignment for tactical weapon applications. In: Proceedings of AIAA guidance, navigation and control conference, Boston; 1989.
[7] Scherzinger BM. Inertial navigator error models for large heading uncertainty. In: Position location and navigation symposium 1996. IEEE; 1996. p. 477–84.
[8] Dmitriyev SP, Stepanov OA, Shepel SV. Nonlinear filtering methods application in INS alignment. IEEE Trans Aerospace Electron Syst 1997;33(January(1)):260–72.
[9] Kong X. INS algorithm using quaternion model for low cost IMU. Robot Auton Syst 2004;46(April(4)):221–46.
[10] Hao Y, Xiong Z, Hu Z. Particle filter for INS In-motion alignment. In: Industrial electronics and applications, 2006 1st IEEE conference on. IEEE; 2006. p. 1–6.
[11] Ali J, Ushaq M. A consistent and robust Kalman filter design for in-motion alignment of inertial navigation system. Measurement 2009;42(May(4)):577–82.
[12] Gao S, Wei W, Zhong Y, Feng Z. Rapid alignment method based on local observability analysis for strapdown inertial navigation system. Acta Astron 2014;94(February(2)):790–8.
[13] Allotta B, Caiti A, Chisci A, et al. unscented Kalman filter based navigation algorithm for autonomous underwater vehicles. Mechatronics 2016;39(November):185–95.
[14] Shin EH, El-Sheimy N. An unscented Kalman filter for in-motion alignment of low–cost IMUs. In: Position location and navigation symposium. IEEE; 2004. p. 273–9.
[15] Hao Y, Xiong Z, Wang W, et al. Rapid transfer alignment based on unscented Kalman filter. In: American control conference. IEEE; 2006. p. 6–12.
[16] Gan Y, Sui L, Wu J, Wang B, et al. An EMD threshold de-noising method for inertial sensors. Measurement 2014;49(March):34–41.
[17] Musavi N, Keighobadi J. Adaptive fuzzy neuro-observer applied to low cost INS/GPS. Appl Soft Comput. 2015;29(Apr):82–94.
[18] Chang L, He H, Qin F. In-Motion initial alignment for odometer-aided strapdown inertial navigation system based on attitude estimation. IEEE Sens J 2016;17(3):766–73.
[19] Nourmohammadi H, Keighobadi J. Fuzzy adaptive integration scheme for low-cost SINS/GPS navigation system. Mech Syst Sig Process 2018;99(January):434–49.
[20] Nourmohammadi H, Keighobadi J. Decentralized INS/GNSS system with MEMS-grade inertial sensors using QR-factorized CKF. IEEE Sens J 2017;17(June(11)):3278–87.
[21] Milanchian H, Keighobadi J, Nourmohammadi H. Magnetic calibration of three-axis strapdown magnetometers for applications in MEMS attitude-heading reference systems. AUT J Model Simul 2015;47(June(1)):55–65.
[22] Stančić R, Graovac S. The integration of strap-down INS and GPS based on adaptive error damping. Robot Auton Syst 2010;58(October(10)):1117–29.
[23] Rogers RM. Applied mathematics in integrated navigation systems. AIAA; 2003.
[24] Titterton D, Weston JL. Strapdown inertial navigation technology. IET; 2004.