18th International Conference on Information Fusion
Washington, DC - July 6-9, 2015

A Hybrid Prediction Method and Its Application in the Distributed Low-cost INS/GPS Integrated Navigation System

Xuemei Wang
Electromechanical Measuring & Controlling Department
Southwest Jiaotong University
Chengdu, China
xmwang2012@swjtu.edu.cn

Jim X. Chen
Computer Science Department
George Mason University
VA, U.S.A.
jchen@gmu.edu

Wenbo Ni
Vehicle Engineering Department
Southwest Jiaotong University
Chengdu, China
nwb@swjtu.edu.cn

Abstract* - *In order to improve the accuracy of INS/GPS integrated navigation system during GPS signals blockage, an effective and low-cost method is to design the corresponding linear or non-linear predictor to predict the position and velocity errors between INS and GPS during GPS blockage and then to correct the results of INS. Based on the distributed data fusion system, a novel hybrid prediction method that combines the radial basis function network (RBFN) and Kalman filter (KF) together was proposed. The predicted value is divided into two parts. One part is the innovation component of KF and the other is the state prediction component of KF. The former is predicted with the designed 6 RBFNs; the latter is predicted with two distributed KFs. Through practical experiments and data processes, it is shown that the proposed hybrid predictor possibly improve the accuracy of INS during GPS blockage.

Keywords: distributed information systems, error correction, GPS outages, inertial navigation, Kalman filter, radial basis function networks.

1 Introduction

With the emergence of the low-cost MEMS gyroscopes and accelerometers, the strapdown inertial navigation system (INS) has been widely used in the land vehicle navigation system [1, 2]. The biggest problem in INS application lies in its lack of a long-term accuracy due to the drift errors of inertial sensors [3, 4]. In the land vehicle navigation application, one of the most effective methods is to adopt the INS/GPS (Global Position System) integrated navigation system [5-7]. The most important advantage of GPS is that it does not have accumulation errors over time and thus can provide the references for INS.

When the land vehicle drove in urban center or in the mountains or tunnels, GPS signals are often blocked or not available, so there is no corresponding references for INS. In this situation, although we can adopt other auxiliary sensors or systems, such as the odometer, the electronic compass, and the digital road map system, to provide the complementary or redundant information, an effective and low-cost method is to design the corresponding linear or non-linear predictors to predict the INS position errors or/and velocity errors during GPS blockage or outages [8-16]. The predictor is actually a dynamic system, which includes the parameters that are needed to be tuned. When GPS signals are normal, the predictor is trained to memorize the dynamic characteristics of the system through the recursive updates of the parameters. Once GPS signals are not available, the predictor can be used to predict the position or/and velocity errors between INS and GPS, which can further be used to correct the results of INS and thus improve the accuracy of the integrated navigation system during GPS blockage or outages.

The researches on the predictors were mainly focused on the choices of the architecture and the training algorithm to meet the needs of fast convergence, high accuracy and flexible suitability to the mobility of the vehicle. Naser EI-sheimy and his research group in Canada researched several kinds of artificial neuron networks (ANN) predictors [8-15]. Reference [8] studied a recursive least-squares lattice (RLSL) predictor, which weighted coefficients can be updated recursively at each time period and is suited for the real-time implementation. The performance of RLSL is only better in relatively lower vehicle dynamic situations but degrades in higher vehicle dynamic situations. Reference [9] studied a multi-layer perception (MLP) network to predict INS latitude and longitude errors. Regarding the limit of the real-time capability of MLP ANN, references [10-12] suggested an alternative ANN architecture——radial basis function network (RBFN) to predict the position errors and/or velocity errors of INS. The RBFN generally has simpler architecture and faster training procedure than the MLP ANN. Furthermore references [13-15] studied an adaptive neuron-fuzzy inference system (ANFIS) network. Generally the increased accuracy usually means using a higher order in the model or a more complicated architecture, which often results in more complicated calculation and lower training speed. References [13, 16] studied a hybrid prediction model, which combined the ANN and KF. This hybrid method made use of the priori dynamic characteristics buried in the KF's state-space equation and achieved higher accuracy with the tactical grade inertial measurement unit (IMU), differential GPS and position and orientation system (POS) [16]. However, making use of POS inevitably increases the cost of the whole navigation system and it is not suitable for the low-cost land vehicle navigation system. What's more, their RBFN inputs are the accelerations and the angular rates measured by inertial sensors, which often include a large amount of measurement noises. The noises make the RBFN training procedure become much more difficult since more neurons may be needed and the corresponding training time inevitably increases accordingly, and what is worse is that it may be divergent.

In practical applications, we should balance between the accuracy, the real-time capability, and the suitability. The main research aim of the paper is to explore a low-cost loosely coupled INS/GPS integrated navigation system that can achieve better balance between the accuracy, the real-time capability, and the suitability even during the periods of GPS blockage or outages. Based on the distributed data fusion method, this paper analyzes and compares the characteristics of the RBFN and a kind of linear neuron network with the time delay at its input (TDLN). Accordingly we propose a distributed low-cost loosely coupled GPS/INS integrated navigation system with the ANN and KF combined prediction method. According to the experiments and data processing results, we have conducted the detailed analyses and comparisons and then drawn a conclusion finally.

2 The design of the predictors

2.1 The distributed data fusion system

As mentioned earlier, the errors of INS often accumulate with time due to the drift errors of inertial sensors. We can make use of the complementary and redundant data provided by GPS to undermine their effects. Although the classical KF exists limits in terms of non-Gaussian and non-linear problems and can only get the corresponding suboptimal estimations in many practical situations, it is still one of the most popular fusion algorithms due to its simplicity and ease of implementation. Particle filter (PF) provides an alternative for KF when dealing with non-Gaussian and non-linear issues. However, PF's computing load is heavy since it may require a large number of random samples to estimate the desired posterior probability density [17]. Its computation burden increases exponentially with dimensionality of the state space. Indeed, PF is not preferable to our INS/GPS integrated navigation system, which includes 15 state variables.

There are two data fusion architectures that are the centralized fusion (also called data level fusion) and the distributed fusion (also called state vectors level fusion) respectively [6]. Theoretically, the centralized fusion approach is the most accurate approach to fuse data, assuming that data association and correlation can be performed correctly [6]. The distributed fusion is not as accurate as the centralized fusion since there is often an information loss on sensors data level fusion so as to make the state vector and its associated covariance matrix only express the quality of the original data approximately. However the distributed fusion reduces the computing load by the use of the lower dimensional matrices and provides the possibility of parallel processing to improve the real-time performance. What's more, based on an appropriate hardware configuration it can also provide the fault tolerance to improve the reliability of the system. Therefore, we divide the observations, which are the difference between the results of INS and signals from GPS, into two groups: one group is the position observations $\delta\boldsymbol{P}=\boldsymbol{P}_{\mathrm{INS}}-\boldsymbol{P}_{\mathrm{GPS}}=(\delta\lambda, \delta L, \delta h)^{\mathrm{T}}$, including the longitude error $\delta\lambda$, the latitude error δL, and the altitude error δh; the other is the velocity observations $\delta\boldsymbol{V}=\boldsymbol{V}_{\mathrm{INS}}-\boldsymbol{V}_{\mathrm{GPS}}=(\delta v_x, \delta v_y, \delta v_z)^{\mathrm{T}}$, including the east-velocity error δv_x, the north-velocity error δv_y, and the up-velocity error δv_z. Separately aided by the two groups' observations, we design a distributed loosely coupled INS/GPS integrated navigation system as shown in Figure 1, in which $\boldsymbol{V}_{\mathrm{INS}}$, $\boldsymbol{P}_{\mathrm{INS}}$, and $\boldsymbol{A}_{\mathrm{INS}}$ denote the INS velocity, position, and attitude respectively; $\delta\hat{\boldsymbol{V}}$, $\delta\hat{\boldsymbol{P}}$, and $\delta\hat{\boldsymbol{A}}$ are the optimal state estimations through two stages data fusion and can be used to correct the INS results; $\boldsymbol{V}_{\mathrm{INS\_C}}$, $\boldsymbol{P}_{\mathrm{INS\_C}}$, and $\boldsymbol{A}_{\mathrm{INS\_C}}$ denote the corrected INS velocity, position, and attitude.

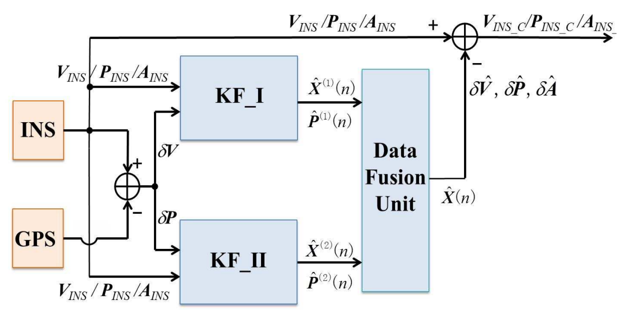


Figure 1. The distributed loosely coupled INS/GPS integrated navigation system.

The state-space equation of the distributed INS/GPS integrated navigation system is expressed as:

$$\boldsymbol{X}(n) = \boldsymbol{F}(n, n-1)\boldsymbol{X}(n-1) + \boldsymbol{\Gamma}(n)\omega(n). \quad (1)$$

Here $\boldsymbol{X}(n)$ is the state vector including 15 state variables, which are the longitude error δL, the latitude error $\delta\lambda$, the altitude error δh, the east-velocity error δv_x, the north-velocity error δv_y, the up-velocity error δv_z, the pitch error $\delta\gamma$, the roll error $\delta\theta$, the yaw error $\delta\psi$, and three direction drift errors of gyroscopes and three direction bias errors of accelerometers, respectively. $\boldsymbol{F}(n,n\text{-}1)$ is the state transition matrix from the previous time point n-1 to the current time point n. $\boldsymbol{\Gamma}(n)$ is the input matrix of the process noise $\omega(n)$. The process noise $\omega(n)$ is assumed to be a zero mean white noise stochastic process.

The observation equation can be expressed as follow:

$$\boldsymbol{z}^{(i)}(n) = \boldsymbol{H}^{(i)}(n)\boldsymbol{X}(n) + v(n). \tag{2}$$

Here i equals to 1 or 2, which corresponds to KF_I or KF_II in Figure 1 respectively. $\boldsymbol{z}^{(1)}(n)$ is the observation vector of KF_I , and $\boldsymbol{z}^{(1)}(n)=\delta\boldsymbol{V}=(\delta v_x, \delta v_y, \delta v_z)^{\mathrm{T}}$. $\boldsymbol{H}^{(1)}(n)$ is the output matrix of KF_I, and

$$\boldsymbol{H}^{(1)}(n) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}_{3\times 15}.$$

$\boldsymbol{R}^{(1)}$ is the observation covariance matrix of KF_I, which is decided by the accuracy of GPS velocity signals. Let's assume that these observations are independent of each other, and

$$\boldsymbol{R}^{(1)} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix}.$$

Similarly, the observation vector $\boldsymbol{z}^{(2)}(n)$, the output matrix $\boldsymbol{H}^{(2)}(n)$, and the observation covariance matrix $\boldsymbol{R}^{(2)}$ of KF_II are expressed as:

$$\boldsymbol{z}^{(2)}(n)=\delta\boldsymbol{P}=(\delta\lambda, \delta L, \delta h)^{\mathrm{T}},$$

$$\boldsymbol{H}^{(2)}(n) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{3\times 15},$$

$$\boldsymbol{R}^{(2)} = \begin{bmatrix} \sigma_\lambda^2 & 0 & 0 \\ 0 & \sigma_L^2 & 0 \\ 0 & 0 & \sigma_h^2 \end{bmatrix}.$$

$\boldsymbol{R}^{(2)}$ is decided by the accuracy of GPS position signals, which are also assumed the mutual independent variables. On the least mean square (LMS) sense we can get the corresponding global optimal fusion result as follows [18]:

$$\hat{\boldsymbol{X}}(n) = \hat{\boldsymbol{P}}(n)\sum_{i=1}^{2}[\hat{\boldsymbol{P}}^{(i)}(n)]^{-1}\hat{\boldsymbol{X}}^{(i)}(n) \tag{3}$$

$$\hat{\boldsymbol{P}}(n) = [\sum_{i=1}^{2}[\hat{\boldsymbol{P}}^{(i)}(n)]^{-1}]^{-1} \tag{4}$$

Here $\hat{\boldsymbol{X}}^{(1)}(n)$ and $\hat{\boldsymbol{X}}^{(2)}(n)$ denote the local state estimations of the two KFs respectively; $\hat{\boldsymbol{P}}^{(1)}(n)$ and $\hat{\boldsymbol{P}}^{(2)}(n)$ denote the corresponding covariance matrices of state estimations of the two KFs respectively; $\hat{\boldsymbol{X}}(n)$ is the global optimal state estimation; $\hat{\boldsymbol{P}}(n)$ is the corresponding covariance matrix.

2.2 The RBFN predictors

The RBFN is highly nonlinear and can be used to model complex input-output mapping [19]. Compared to other nonlinear networks, the RBFN is relatively simpler and often takes much less time for training, which is important for the real-time application. However the RBFN often requires more neurons than the standard feedforward backpropagation network. The most common used radial basis function in the RBFN is the Gaussian function. A typical FBFN is a three-layer network with an input layer, an intermediate layer of Gaussian units and an output layer of linear summation units [19]. Figure 2 is the RBFN with single input and single output.

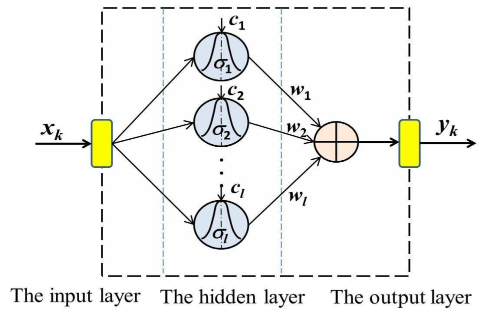


Figure 2. The RBFN with single input and single output

Let's assume that x_k is the input of the RBFN at time point k; y_k is the output of the RBFN at time point k; u_i is the output of the ith Gaussian neuron, which is expressed as:

$$u_i = \exp[-\frac{1}{2}(\frac{x_k - c_i}{\sigma_i})^2],\ i=1, 2, \ldots, l. \tag{5}$$

Here, l *is* the number of Gaussian neurons; c_i is the center parameter of the ith Gaussian neuron; σ_i is the standard derivation parameter of the ith Gaussian neuron. It can be seen that the output of each Gaussian neuron depends on how close the input x_k is to each neuron's center parameter c_i. The standard derivation parameter σ_i in each neuron determines the width of an area in the input space to which each neuron responds. The final output y_k of the RBFN is expressed as:

$$y_k = \sum_{i=1}^{l} w_i \exp[-\frac{1}{2}(\frac{x_k - c_i}{\sigma_i})^2] \tag{6}$$

Here, w_i (i=1~l) is the weighted parameters of the linear combiner.

For the RBFN, the parameters that need to be tuned are c_i, σ_i and w_i. The training procedure of the RBFN can be divided into two stages. For the first stage a non-supervision learning rule called K-means clustering algorithm can be adopted; for the second stage the gradient descent method on the LMS sense is usually used to adjust the weighted parameters w_i [19].

In the prediction application during GPS signals blockage or outages, we design 6 RBFNs with single input and single output as above. The inputs are the longitude, the latitude, the altitude, the east-velocity, the north-velocity, and the up-velocity of INS respectively for each RBFN. The corresponding outputs of 6 RBFNs are the errors between INS positions & velocities and GPS positions & velocities

Table 1. Comparisons of the TDLN and the RBFN predictors.

Order number	Input-output signals			TDLN		RBFN	
	SNR (dB)	f_1/f_2	Phase difference $\Delta\varphi$	Training performance (MSE)	Prediction Performance (MSE)	Training performance (MSE)	Prediction Performance (MSE)
1	40	1/1	0	2.4860×10^{-7}	1.5841×10^{-6}	8.5821×10^{-5}	8.8548×10^{-5}
2	30	1/1	0	3.5216×10^{-6}	1.4652×10^{-5}	8.2258×10^{-4}	1.2500×10^{-3}
3	20	1/1	0	2.3534×10^{-5}	1.2498×10^{-4}	7.4680×10^{-3}	8.9000×10^{-3}
4	40	1/2	$\pi/4$	0.3783	0.8741	6.6824×10^{-4}	7.7128×10^{-4}

SNR is just for the RBFN input signals. The training performances of the TDLN are gained at 3 ~ 6 epoch; the training performances of the RBFN are gained at 500 epoch. Simulations are carried out with the help of MATLAB Neural Network Toolbox.

respectively. The reason that we design 6 RBFNs with single input and single output lies in two considerations: on the one hand, since the different position or velocity signals often have the absolutely different characteristics, they should be modeled and characterized separately; on the other hand, 6 RBFNs could be implemented in parallel to meet the requirement of the real-time application. A simpler RBFN with single input and single output obviously takes much less time for training than a comparatively complicated RBFN with more inputs and outputs.

During our researches, we also found that the RBFN was sensitive to the input noises, which make the corresponding prediction errors increase. Let's assume that the RBFN input is a sinusoidal signal with a_m magnitudes, f_1 Hz frequencies, and φ_1 initial phases and the input includes an additive random noise u, which is the Gaussian white noise with zero mean. The target output of the RBFN is another sinusoidal signal with the same magnitudes, f_2 Hz frequencies, and φ_2 initial phases. One thousand groups of data are simulated. The first half groups are used for training the RBFN and the second half groups are used for verification and prediction. In the different signal to noise ratios (SNR) and the different f_1 to f_2 ratios (f_1/f_2), we carry out the simulations and get the prediction outputs and the corresponding prediction errors, which are shown in Table 1. At the same time we also give out the prediction errors of a linear neural network with 200 steps time delay at the input (simply called TDLN). Compared with the outputs of the TDLN, we can see that the designed RBFN is much more susceptible to noises than the TDLN. With the SNR decline in input signals, the RBFN prediction errors increase greatly. It is known that the measurement signals of inertial sensors often include enormous random noises due to the drift errors of inertial sensors and some other environmental related factors. Although we can carry out filtering process to reduce the noises level partly, the measurement signals coming from inertial sensors still includes much more fluctuations compared to the results of INS. The results of INS are the positions, velocities, and attitudes of the vehicle obtained by the integral calculation. They are often much smoother than the signals directly from inertial sensors. Therefore we use the positions and velocities of INS as the 6 RBFNs inputs instead of the angular rate or acceleration signals.

When there is a linear relationship between the inputs and the outputs at f_1/f_2 =1 and $\Delta\varphi$ =0, the corresponding linear network that has not only fast training speed, but also high prediction accuracy, is a better choice. However when there is a nonlinear relationship between the inputs and the outputs, for example at f_1/f_2 =1/2 and $\Delta\varphi$ =π/4, the RBFN is obviously better than the linear network.

2.3 The hybrid predictor combined the KF with the RBFN

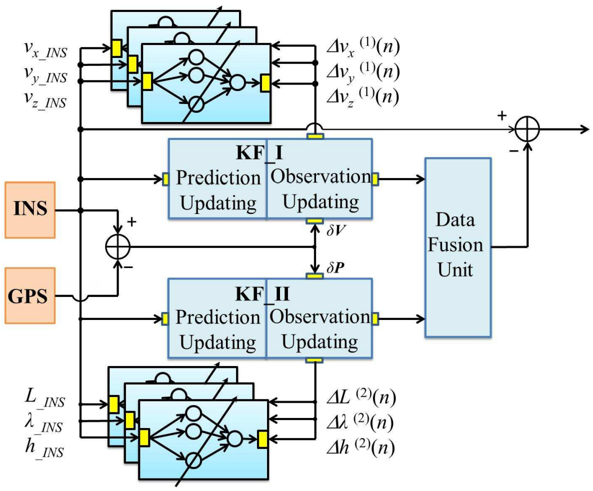
No matter what it is linear or non-linear, the predictor designed above is modelled according to the input and the output data and does not make use of any priori motion and dynamic characteristics of the navigation system at all. Thus in high vehicle mobility situations or over long term of GPS blockage or outages, the trained predictor in the last time period may be totally not suited for the following time period. The suitability of the model is degraded and we often cannot obtain the expected accuracy at all. As we have known, the state-space equation of KF includes a large amount of priori information about the motion of the body and reflects the dynamic characteristics of the navigation system. How do we use it? The recursive calculation procedure of KF includes the prediction update and the observation update two periods. The prediction update process of KF is to get the state prediction at the current time point n according to the state estimation at the last time point n-1 [18]. The prediction update process is up to the corresponding state transition matrix $\boldsymbol{F}(n,n-1)$, which is derived from the equations of inertial navigation calculation and buries a large amount of information on the dynamic characteristics of INS. During the recursive process the state transition matrix $\boldsymbol{F}(n,n-1)$ should continuously acquire the position, the velocity, and the attitude of the body from INS to update itself at each period. During GPS blockage or outages, we can only make use of the latest updated observations from GPS. Although the Kalman filtering process is not perfect because of the lack of the observations updating from GPS, it still can provide the corresponding dynamic information of INS partly. So we combine the KF and the RBFN to design a kind of hybrid predictor. We divide the state estimations $\hat{\boldsymbol{X}}^{(i)}(n)$ of KF, which reflects the differences between the results of INS and the observations of GPS, into two components. One is the state prediction component $\hat{\boldsymbol{X}}^{(i)}(n \mid n-1)$ from the prediction update period;

the other is the innovation component $\Delta\hat{\boldsymbol{X}}^{(i)}(n)$ from the observation update period. And there are:

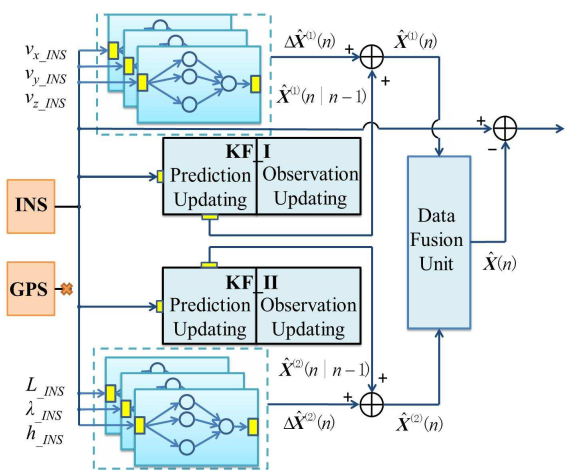
$$\hat{\boldsymbol{X}}^{(i)}(n) = \hat{\boldsymbol{X}}^{(i)}(n \mid n-1) + \Delta\hat{\boldsymbol{X}}^{(i)}(n) . \tag{7}$$

Here the state estimations are particularly referred to the position errors and velocity errors between INS and GPS. After the equation (7) is expanded, we can get:

$$\begin{aligned} &\delta v_x^{(1)}(n) = \delta v_x^{(1)}(n|n-1) + \Delta v_x^{(1)}(n), \\ &\delta v_y^{(1)}(n) = \delta v_y^{(1)}(n|n-1) + \Delta v_y^{(1)}(n), \\ &\delta v_z^{(1)}(n) = \delta v_z^{(1)}(n|n-1) + \Delta v_z^{(1)}(n), \\ &\delta L^{(2)}(n) = \delta L^{(2)}(n|n-1) + \Delta L^{(2)}(n), \\ &\delta\lambda^{(2)}(n) = \delta\lambda^{(2)}(n|n-1) + \Delta\lambda^{(2)}(n), \\ &\delta h^{(2)}(n) = \delta h^{(2)}(n|n-1) + \Delta h^{(2)}(n). \end{aligned} \tag{8}$$



(a) Training procedure during GPS normal



(b)Predicting procedure during GPS outages

Figure 3. Distributed INS/GPS integrated navigation system based on the hybrid predictor

When GPS signals are available, the distributed KF data fusion system works normally and 6 RBFNs are being trained in real-time as shown in Figure 3(a). During the training stage, the inputs of 6 RBFNs are the longitude $L_{\_INS}$, the latitude $\lambda_{\_INS}$, the altitude $h_{\_INS}$, the east-velocity v_{x}_{INS}, the north-velocity v_{y}_{INS}, and the up-velocity v_{z}_{INS} of the body from INS respectively. The corresponding outputs of them are the innovation components $\Delta\hat{\boldsymbol{X}}^{(i)}(n)$ from the observation update periods of the two KFs. Through the real-time training, 6 RBFNs build up the non-linear mappings between the results of INS and the innovation components of KFs. When GPS signals are not available, although the distributed KF data fusion system cannot work completely normally, it still can approximately provide the state predictions $\hat{\boldsymbol{X}}^{(i)}(n \mid n-1)$, which make use of the motion and dynamic characteristics buried in the state transition matrix $\boldsymbol{F}(n, n\text{-}1)$. At this situation, 6 RBFNs work as the predictors and provide the predictions of the innovation components $\Delta\hat{\boldsymbol{X}}^{(i)}(n)$ of KFs. Then the state predictions $\hat{\boldsymbol{X}}^{(i)}(n|n-1)$ from the two KFs and the innovation predictions $\Delta\hat{\boldsymbol{X}}^{(i)}(n)$ form 6 RBFNs are respectively summed together to form the local state estimations $\hat{\boldsymbol{X}}^{(i)}(n)$ as shown in Figure 3(b). Then we can further carry on the optimal data fusion to get the optimal state estimations $\hat{\boldsymbol{X}}(n)$ and finally correct the results of INS with them.

3 Experiments and data analysis

3.1 Experimental method and setup

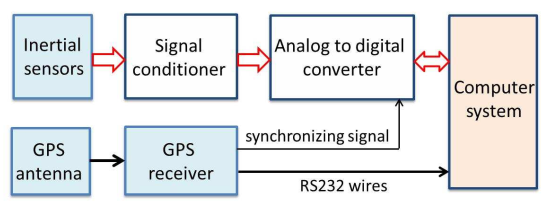


Figure4. The block diagram of the experimental setup

We set up the corresponding experimental setup which block diagram is shown in Figure 4. It is composed of the inertial sensors, data acquisition board, OEMStar low cost single-frequency GPS receiver from NovAtel and data processing computer. The inertial sensors include three low-cost MEMS-based gyroscopes and three low-cost MEMS-based accelerometers. The performance indexes of inertial sensors are listed in Table 2. The experimental setup was mounted on the vehicle that moved along the track on the campus. The inertial sensors data are conditioned, sampled and then processed in the computer. GPS signals are received by the receiver board and are transmitted to the computer through RS232 serial interface. It is very crucial to keep the synchronization between inertial sensors' measured data and GPS received signals. We utilized 1 PPS (one pulse per second) signal from GPS receiver to be synchronizing signal to the data acquisition board. The sampling frequency of inertial sensors measured data is 500Hz and the updating frequency of GPS signals is 1Hz. In the data process, we assumed that the GPS signals maintained the latest receiving values during every updating period. During the whole experiment procedure, the numbers of satellites that could be observed were more than 4 and the GPS signals were very well.

Table 2. The performance indexes of inertial sensors.

Indexes	Accelerometers	Gyroscopes
Voltage supply	±12V	±12V
Supply current	<20mA	<30mA
Range	±2g	±20 °/s
Resolution	0.001g	< 0.01°/s
Sensitivity	2000±30mV/g	500mV/°/s
Nonlinearity	0.5%FS	<0.1%FS
Band Width	400Hz	60Hz
Noise	5（$\mu g/\sqrt{Hz}$）	
Zero bias	0±0.2V	±0.5 °/s
Sensitivity temperature drift		0.5%（max）
Bias temperature drift		±0.2 °/s
Bias stability		±0.02 °/s

3.2 Data processes and analysis

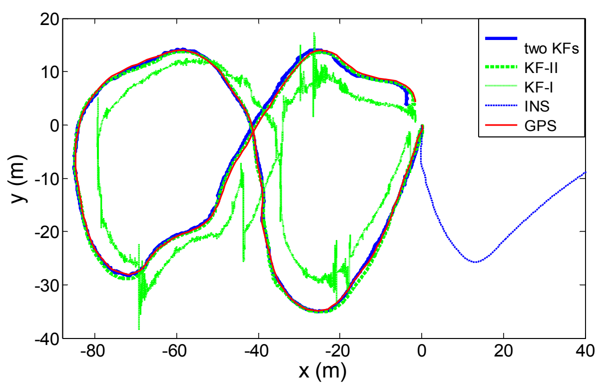
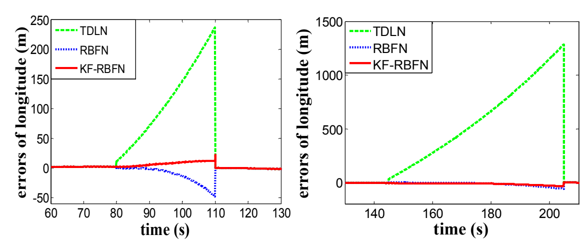


Figure 5. The trajectories obtained with the different fusion methods : the trajectory in green dotted lines is obtained from KF_I; the trajectory in green dashed lines is from KF_II; the trajectory in blue solid lines is from the two KFs fusion; the trajectory in red solid lines is the GPS reference; the trajectory in blue dashed lines is from the calculation of INS.

We used GPS signals as the references and processed the experiment data based on KF data fusion algorithm to get the 2D trajectories of the vehicle as shown in Figure 5. In Figure 5, the trajectory observed by GPS was a '∞' shape denoted in red solid lines. The computation results of INS denoted in blue dash lines exist a huge deviation from GPS observations due to the drift errors of inertial sensors and some environmental factors. In Figure 5 there are also results of two separate KFs which observations are GPS positions and GPS velocities respectively. We can see that with the help of the observations from GPS the navigation accuracy of INS is improved obviously. The corresponding position state estimations of KF_I with GPS velocities as the observations are not better than that of KF_II with GPS positions as the observations. If we further fuse the results from two separate KFs, we can get the more accurate results that are also shown in Figure 5 in blue solid lines, which achieves about 1.2m root mean square (RMS) error accuracy relative to GPS positions.

To verify the effectiveness of the designed hybrid predictors during GPS blockage or outages, we intentionally cut off GPS signals at 30 seconds and 60 seconds time intervals respectively. From Figure 6 to Figure 11, we provide the corresponding errors of the longitude, the latitude, the altitude, the east-velocity, the north-velocity and the up-velocity when they are corrected with the different predictors during GPS outages. We can see that all positions and velocities during GPS outages are improved to some extent with the help of these predictors. Generally, the performance of the designed KF-RBFN hybrid predictors is superior to that of the RBFN predictors or the TDLN predictors alone, which is proved in Figure 6, Figure 8, Figure 9 and Figure 11. What's more, we can see that the performance of the KF-RBFN hybrid predictors is particularly outstanding in the correction of the altitude and the up-velocity. But there are also some exceptions, which can be seen in Figure 7 and Figure 10. The accuracy of the latitude corrected with the KF-RBFN hybrid predictors is equivalent to that with the RBFN predictors or the TDLN predictors alone. The accuracy of the north-velocity corrected with the KF-RBFN hybrid predictors is a little less than that with the TDLN or the RBFN predictors. From Figure 6, Figure 9, and Figure 11, we can also see that in most instances the performance of the RBFN predictors is better than that of the TDLN predictors. The performance of the predictor is mainly up to its architecture and algorithm. Moreover it is also closely related to the data's variation characteristics, which are very complicated in practical situations. Table 3 gives out the RMS errors of the positions and the velocities during 30 seconds and 60 seconds GPS outages intervals with the different predictors. We can see that the accuracy of the predictors is declining with the increase of GPS outages time interval.

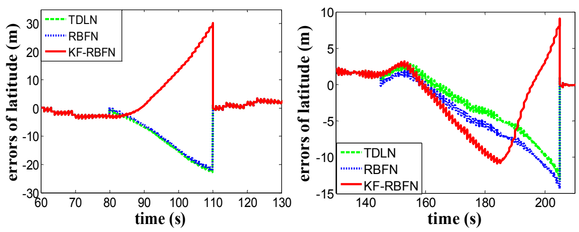


(a) 30s outage interval (b) 60s outage interval

Figure 6. Errors of the longitude during GPS outages.

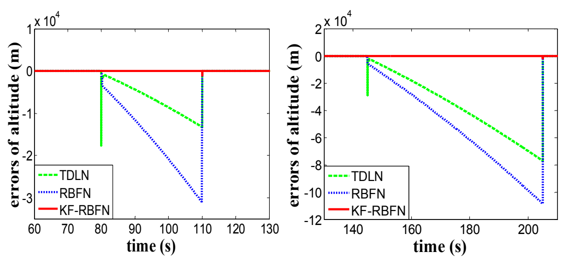
Table 3. RMS errors of positions and velocities with the different predictors during GPS outages.

Items		TDLN Predictors		RBFN Predictors		Hybrid Predictors	
		30s outage interval	60s outage interval	30s outage interval	60s outage interval	30s outage interval	60s outage interval
Position RMS Errors(m)	Longitude	48.8457	368.5379	6.7454	12.0465	2.9495	8.8609
	Latitude	4.8964	2.7340	4.7034	3.3548	4.9846	3.0110
	Altitude	2.837e+3	2.2941e+4	6.8774e+3	3.2891e+4	0.3719	2.5718
Velocity RMS Errors(m/s)	East	0.5132	2.7722	0.1886	1.5799	0.1162	0.2351
	North	0.2328	0.1926	0.1615	0.2314	0.5534	0.4023
	Up	133.4138	248.3301	30.4365	158.1367	0.0190	0.0500



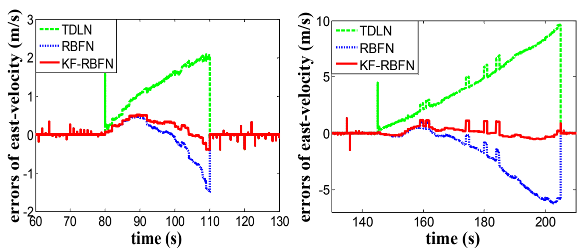
(a) 30s outage interval (b) 60s outage interval

Figure 7. Errors of the latitude during GPS outages.



(a) 30s outage interval (b) 60s outage interval

Figure 8. Errors of the altitude during GPS outages.

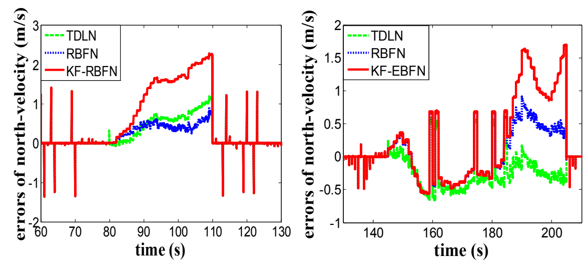


(a) 30s outage interval (b) 60s outage interval

Figure 9. Errors of the east-velocity during GPS outages.

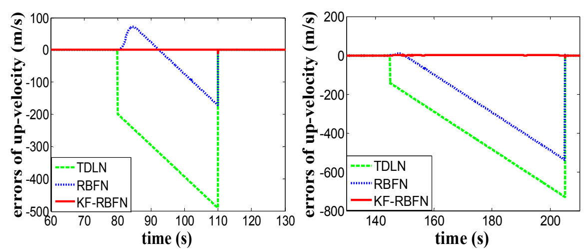
4 Conclusion

In the INS/GPS integrated navigation system application GPS signals are often blocked and unavailable in the urban center or in the mountains or tunnels. In order to improve the accuracy of INS in such situation, an effective and low-cost method is to design the corresponding linear or non-linear predictors to predict the errors between INS and GPS and then use the predicted errors to correct the results of INS. We analyze and compare the performances of the RBFN and the TDLN. For the purpose of achieving better balance between the accuracy, the real-time capability and the suitability for nonlinearity, we choose and design 6 RBFNs predictors. However, the classical predictor is modeled according to the input and the output data, which does not make use of any priori dynamic characteristics of the navigation system at all so that in many situations the suitability of the model is degraded and cannot realize the expected accuracy. In view of the state transition matrix of KF, which includes a large amount of information about the dynamic characteristics of INS, based on the distributed INS/GPS integrated navigation system we propose a hybrid prediction method that combines the designed 6 RBFNs and two KFs together. 6RBFNs are trained and used to predict the innovation components of the state estimations. Two KFs are used to provide the state prediction components from the last time point n-1 to the current time point n. Two parts are summed together to form the expected predictions to correct the results of INS. Through the practical experiments and data processing, it shows that the designed hybrid predictors improve the accuracy of INS during GPS



(a) 30s outage interval (b) 60s outage interval

Figure 10. Errors of the north-velocity during GPS outages.



(a) 30s outage interval (b) 60s outage interval

Figure 11. Errors of the up-velocity during GPS outages.

signals blockage or outages and achieve about 9.3m 2D position errors accuracy and 0.45m/s velocity errors accuracy with the low-cost and low grade inertial sensors and GPS receiver. At the same time, we also see that any method has its limits. Although in most cases the designed hybrid predictor is better than the RBFN and the TDLN, there are also some exceptions because of the complicated signals scenarios.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities of China under Project SWJTU11CX021 and the State Scholarship Fund of China under Project 201307005026.

References

[1] J. H. Wang, Y. Gao, "Land Vehicle Dynamics-Aided Inertial Navigation," *IEEE Trans. Aerosp. and Electron. Syst.*, vol.46, no.4, pp.1638-1653, Oct. 2010.

[2] H. Liu, S. Nassar, N. El-Sheimy, "Two-Filter Smoothing for Accurate INS/GPS Land-Vehicle Navigation in Urban Centers," *IEEE Trans. Veh. Technol.,* vol.59, no.9, pp.4256-4267, Nov. 2010.

[3] D. H. Titterton, and J. L. Weston, *Strapdown Inertial Navigation Technology 2nd ed.*, Institution of Engineering and Technology, Stevenage, 2004 .

[4] J. Georgy, A. Noureldin, M. J. Korenberg, M. M. Bayoumi, "Modeling the Stochastic Drift of a MEMS-Based Gyroscope in Gyro/Odometer/GPS Integrated Navigation," *IEEE Trans. Intell. Transp. Syst.,* vol.11, no.4, pp.856-872, Dec. 2010.

[5] D. L. Hall, J. Llinas, "An introduction to multisensor data fusion," Proceedings of the IEEE, vol.85, no.1, pp.6-23, Jan. 1997.

[6] K. Bahador, K. Alaa, O. K. Fakhreddine, N. R. Saiedeh, "Multisensor data fusion: A review of the state-of-the-art," Information Fusion, vol.14, no.1, pp.28-44,Jan. 2013.

[7] M. S. Grewal, L. R. Weill, and A. P. Andrew, Global Positioning, Inertial Navigation & Integration (Second Edition), John Wiley & Sons, New York, 2007.

[8] M. El-Gizawy, A. Noureldin, and N. El-Sheimy, "A reliable modeless mobile multi-sensor integration technique based on RLS-lattice," Meas. Sci. Technol., vol.17, no. 1, pp. 51-61, Jan. 2006.

[9] A. Noureldin, A. Osman, and N. El-Sheimy, "A neuro-wavelet method for multi-sensor system integration for vehicular navigation," Meas. Sci. Technol., vol.15, no.2, pp.404-412, Feb. 2004.

[10] R. Sharaf, A. Noureldin, A. Osman, and N. El-Sheimy, "Online INS/GPS integration with a radial basis function neural network," IEEE A&E Syst. Mag., vol.20, no.3, pp.8-14, Mar. 2005.

[11] A. Noureldin, R. Sharaf, A. Osman, and N. El-Sheimy, "INS/GPS data fusion technique utilizing radial basis functions neural networks," Position Location and Navigation Symposium 2004, PLANS 2004, 26-29 Apr. 2004, pp.280-284.

[12] L. Semeniuk, and A. Noureldin, "Bridging GPS outages using neural network estimates of INS position and velocity errors," Meas. Sci. Technol. vol.17, no.10, pp.2783–2798, Oct. 2006.

[13] W. Abdel-Hamid, A. Noureldin, and N. El-Sheimy, "Adaptive Fuzzy Prediction of Low-Cost Inertial-Based Positioning Errors," IEEE Trans. fuzzy syst., vol.15, no.3, pp.519-529, Jun. 2007.

[14] A. Noureldin, A. El-Shafie, and N. El-Sheimy, "Adaptive neuro-fuzzy module for inertial navigation system/global positioning system integration utilising position and velocity updates with real-time cross-validation," Radar, Sonar & Navigation, IET , vol.1, no.5, pp.388-396, Oct. 2007.

[15] M. El-Gizawy, A. Noureldin, and N. El-Sheimy, "Integrated recursive least square lattice and neuro-fuzzy modules for mobile multi-sensor data fusion," Electronics, Circuits and Systems, 2005, 12th IEEE Int. Conf. on ICECS, Gammarth and 11-14 Dec. 2005, pp1-4.

[16] L. Z. T. Chen, J. C. Fang, "A Hybrid Prediction Method for Bridging GPS Outages in High-Precision POS Application," IEEE Trans. Instrum. Meas., vol.63, no.6, pp.1656-1665, Jun 2014.

[17] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Trans. Signal Processing, vol.50, no.2, pp.174-188, Feb. 2002.

[18] Z. S. He, and W. Xia, Modern digital signal processing and application, Tsinghua Univ. press, Beijing(China), 2010.

[19] Paulo S. R. Diniz, Adaptive Filtering: Algorithms and Practical Implementation, Springer Science Business Media, New York, 2013.