

Transport and Telecommunication, 2017, volume 18, no. 2, 146–154
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia
DOI 10.1515/tjt-2017-0014

GNSS POSITIONING PERFORMANCE ANALYSIS USING PSO-RBF ESTIMATION MODEL

Meriem Jgouta¹, Benayad Nsiri²

¹*University Hassan II, Faculty of Sciences
Casablanca, Morocco, Km 8 Eljadida road, P.B 5366, Maarif 20100*

¹ *00212667948795, mariam.jgouta@gmail.com*

² *00212663769740, benayad.nsiri@enst-bretagne.fr*

Positioning solutions need to be more precise and available. The most frequent method used nowadays includes a GPS receiver, sometimes supported by other sensors. Generally, GPS and GNSS suffer from spreading perturbations that produce biases on pseudo-range measurements. With a view to optimize the use of the satellites received, we offer a positioning algorithm with pseudo range error modelling with the contribution of an appropriate filtering process. Extended Kalman Filter, The Rao-Blackwellized filter are among the most widely used algorithms to predict errors and to filter the high frequency noise. This paper describes a new method of estimating the pseudo-range errors based on the PSO-RBF model which achieves an optimal training criterion. This model is appropriate of its method to predict the GPS corrections for accurate positioning, it reduce the positioning errors at high velocities by more than 50% compared to the RLS or EKF methods.

Keywords: GNSS; pseudo-range; filtering; prediction; PSO-RBF

1. Introduction

The Satellite navigation systems have entered the world of transport. Using the American Global Positioning System (GPS) and the traveller information services, tracking containers or fleet management is now possible. These services do not require a high demand for neither availability, nor accuracy nor integrity. The deployment of these services and the imminent arrival of new global coverage systems (Galileo, Beidou or Glonass) encourage the scientific community to develop new services with stronger constraints (Santerre *et al.*, 1995). These competing systems contribute to GPS and will increase availability as well as improve the security and reliability of supply through information on the signal integrity of Galileo. Global Navigation Satellite Systems (GNSS) performance is defined by the parameters of the system (measurements and user processing software), they also rely heavily on signal propagation environment. However, one of the main drawbacks of GNSS is the capacity to reach the Required Navigation Performance in terms of accuracy and availability.

The traditional filtering methods, such as Kalman filter, are used by the GNSS receiver to lessen positioning error. However, the working of these methods can be degraded, because of Non-Line-of-Sight (NLOS) reception. Therefore, it is advantageous to study that relative performance between those algorithms. A comparative analysis on standard estimation algorithms used in GNSS receiver has been shown in (Jgouta and Nsiri, 2015). Mean squared error (MSE) is used as performance measure. Even though the methods cited before in (Jgouta and Nsiri, 2015) are used vastly, in most cases they can't give a precise positioning estimate.

We saw previously that the Rao-Blackwellized Particle Filter (RBPF) reach good results in solving the nonlinear and non-Gaussian problems (Jgouta and Nsiri, 2015). Indeed, RBPF could not just ameliorate the estimate exactness but also diminish the general computational intricacy. Though, the computational burden is even too high for many real-time applications. To get better efficiency than RBPF, the particle swarm optimization (PSO) is carried out to take all the particles to the areas where their probabilities are high in the nonlinear region. Hence, just a few particles are required to participate in the calculation. There are many advantages of using Radial basis function (RBF) networks such as surmounting the problem of choosing the neurons number in their concealed layer, as they are generated with dynamism during the training process to attain the wanted performance.

RBF neural network is a type of feed forward neural network. It is an input layer, a nonlinear hidden layer and a linear output layer. RBF utilizes basis functions in the hidden layer (Leonard and Kramer, 1991) and (Shen *et al.*, 2002). PSO proposed by Kennedy and Eberhart (Cui and Polok, 2005) is a member of the broad category of swarm intelligence methods in order to solve nonlinear programming issues. Training techniques may be formulated as an optimization which includes the network structure

into a set of variables that are used to minimize the prediction error. Furthermore, PSO is used to optimize the RBF neural network parameters in this work.

This paper is organized as follows. First, we will review how to determine the position of a GPS receiver. Next, a new approach is suggested to optimize GPS positioning. The procedure makes use of the RBF and the PSO techniques, resulting in a system called PSO-RBF, which is used within the navigation systems for real-time identification of noise covariance matrices to impede the divergence of the simple Kalman filter. Finally, the simulation results and comparison with other methods are presented.

2. GNSS Measurement Model

2.1. Pseudo range measurements

All GNSS receivers can acquire the position of the satellites that are visible to it from their radio navigation signals. In the case of GPS, the signal carries the navigation message. This contains alternative information required for positioning. This information is the date of transmission of the signal, the clock corrections and ephemeris, an indicator of proper operation of the satellite transmitter and a specific indicator for each satellite, the coefficients necessary for ionospheric correction and coefficients for calculating time Coordinated Universal time (UTC).

The basic principle of the navigation based on the known and constant wave velocity (equivalent to the speed of light). Therefore, the knowledge of signal propagation time between the transmitter and the receiver can determine the distance between the transmitter and this receiver. This time is measured by the cross-correlation between the replica of the Pseudo-Random Noise code (PRN), Clear /Acquisition code (C/A) or Precision code (P), generated by the receiver and the PRN code transmitted by the satellites (Mosavi and Rahemi, 2015). In current propagation conditions, a number of phenomena lead to errors in the pseudo range. According to the literature and in Cartesian coordinates, the pseudo range measured on the signal from satellite i is as follows:

$$P_s = \rho + c(\delta t_r - \delta t_s) + e_s, \tag{1}$$

P_s : pseudo-range measurement, ρ : the true range of the satellite from the user receiver,
 c : the celerity, δt_r and δt_s : the receiver and satellite clock bias respectively,
 e_s : The pseudo-range errors for satellite s .

$$e_s = I_s + T_s + M_s + N, \tag{2}$$

I_s and T_s are the ionospheric and tropospheric errors,
 M_s is the error caused by signal reflections (multipath errors), and N is the noise receiver.

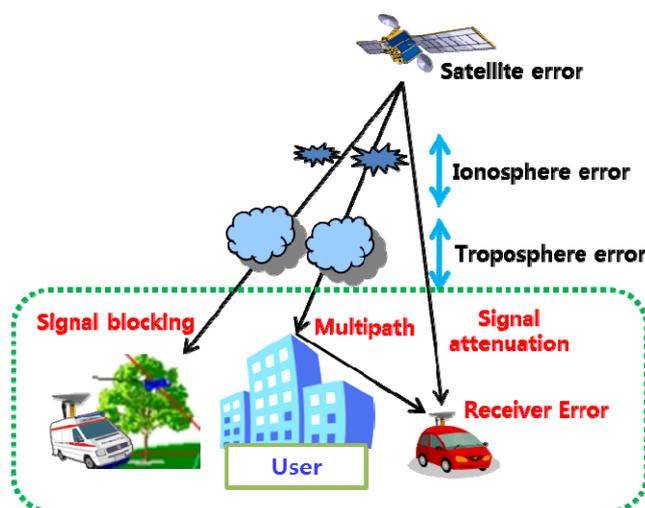


Figure 1. Errors sources of satellite signals (Daehee Won, Precise Positioning, http://smileforday.com/?page_id=84)

2.2. Modelling state

We will then use this modelling to perform matrix calculations to estimate positions (Jgouta and Nsiri, 2015). However, for the first estimators adapted to the GNSS, this will be possible after linearization. Indeed, according to equation (1), the expression of the pseudo-distance is weakly non-linear compared to unknown (x_r, y_r, z_r) .

We set $\Delta t = (\delta t_r - \delta t_s)$ and considering $\rho = f(x_s, y_s, z_s)$ the nonlinear expression of equation (1):

$$f(x_s, y_s, z_s) = \sqrt{(x_r - x_s)^2 + (y_r - y_s)^2 + (z_r - z_s)^2}. \quad (3)$$

To linearize f , we will use the limited development in Taylor series, this development cannot be done only in the locality of a known position (origin or previous position). We need to redefine the position estimate based on the known position as follows:

$$x_r = x_{t_0} + \Delta x_r, \quad y_r = y_{t_0} + \Delta y_r, \quad z_r = z_{t_0} + \Delta z_r. \quad (4)$$

With $(x_{t_0}, y_{t_0}, z_{t_0})$ is the initial position and $(\Delta x_r, \Delta y_r, \Delta z_r)$ is the update of the position relative to the initial position at time t_0 . Using Taylor's series and limiting to the first order the equation (2) and (3) can be developed as below:

$$f(x_r, y_r, z_r) = f(x_{t_0}, y_{t_0}, z_{t_0}) + \frac{\partial f(x_{t_0}, y_{t_0}, z_{t_0})}{\partial x_{t_0}} \Delta x_r + \frac{\partial f(x_{t_0}, y_{t_0}, z_{t_0})}{\partial y_{t_0}} \Delta y_r + \frac{\partial f(x_{t_0}, y_{t_0}, z_{t_0})}{\partial z_{t_0}} \Delta z_r. \quad (5)$$

Developing the partial derivative of the equation (4) and putting $d_{s,0}$ as the approximation of the satellite-receiver distance:

$$d_{s,0} = \sqrt{(x_s - x_0)^2 + (y_s - y_0)^2 + (z_s - z_0)^2}. \quad (6)$$

The expression of the pseudo range becomes:

$$P_s = d_{s,0} - \frac{x_s - x_0}{d_{s,0}} \Delta x_r - \frac{y_s - y_0}{d_{s,0}} \Delta y_r - \frac{z_s - z_0}{d_{s,0}} \Delta z_r + c \Delta t_r. \quad (7)$$

To estimate a position, one pseudo-range measurement is not sufficient. We need a minimum of four different measurements to determine a position in 3D. In practice, the receiver takes into consideration all of the pseudo-measures disposable to compensate for any perturbation of signals or any bad configurations. Therefore, it is necessary to reformulate the equation (7) in matrix form by taking into account all disposable pseudo-ranges. This reformulation is given as follows:

$$\begin{bmatrix} P_1 - d_{1,0} \\ \vdots \\ P_n - d_{n,0} \end{bmatrix} = \begin{bmatrix} -\frac{x_1 - x_0}{d_{1,0}} & -\frac{y_1 - y_0}{d_{1,0}} & -\frac{z_1 - z_0}{d_{1,0}} & c \\ \vdots & \vdots & \vdots & \vdots \\ -\frac{x_n - x_0}{d_{n,0}} & -\frac{y_n - y_0}{d_{n,0}} & -\frac{z_n - z_0}{d_{n,0}} & c \end{bmatrix} \begin{bmatrix} \Delta x_r \\ \Delta y_r \\ \Delta z_r \\ \Delta t_r \end{bmatrix}. \quad (8)$$

To simplify the notation we put $Y_{s_i} = P_{s_i} - d_{s,0_i}$, the matrix representation of (8) is identified as:

$$Y = HX. \quad (9)$$

The equation (9) is the matrix form of the observation equation.

3. Particle Swarm Optimization and Radial Basis Function Network

3.1. The Particle Swarm Optimization Algorithm

The principle of PSO was first proposed by J. Kennedy and R. Eberhart in 1995. PSO is an evolutionary computing algorithm approach inspired by the social behaviour of birds; it is based on repetition and starts with a random matrix as a primary population (Mussi *et al.*, 2009).

A swarm based on a set of particles moving around the search space, each one symbolizes a potential solution (fitness) (Li, J. and Li, B., 2014). There are two parameters for each particle containing positioned velocity of the particle, which are defined by a space vector x_i and a velocity vector v_i , respectively.

The position at which the best fitness p_{best} met by the particle, and the index of the best particle g_{best} in the swarm. Every generation, the position of each particle is updated by adding the velocity to the position vector (Dehuri and Tripathy, 2011). These particles form a pattern in an n-dimensional space and move to the required value. In accordance with the experience from the previous moves, the particles take a decision about how to decide the next move. In every iteration, all particles in the n-dimensional problem space move to an optimum point. The position and velocity of each particle can be changed conferring to the following equations:

$$v_i^{t+1} = wv_i^t + C_1 \times rand \times (p_{best_i}^t - x_i^t) + C_2 \times rand \times (g_{best_i}^t - x_i^t).$$

The positions are based on their movement over a discrete time interval Δt usually set to 1 as follows:

$$x_i^{t+1} = x_i^t + v_i^{t+1} .$$

The parameters C_1 and C_2 are set to positive constant values, which are normally taken as two whereas $rand$ represent uniformly distributed random values in $[0,1]$, and w is called as inertia weight that can be constant (Azami *et al.*, 2013), the inertia weight is employed to control the impact of the previous history of velocities on the current one. Figure 1 shows the flowchart for the PSO algorithm.

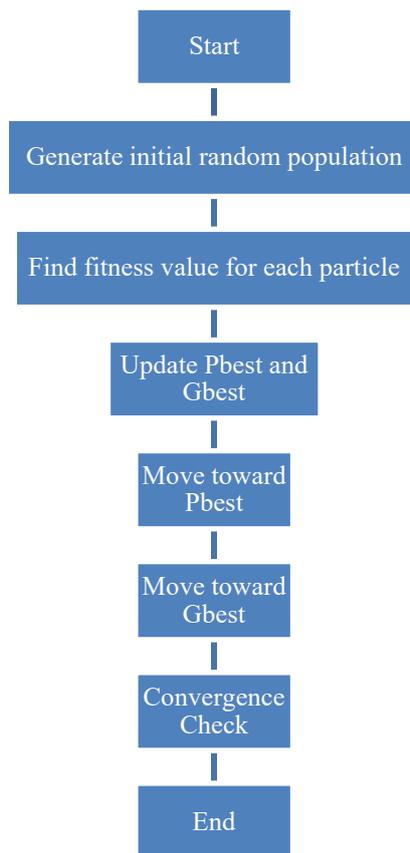


Figure 2. Flowchart for the PSO algorithm

3.2. The Radial Basis Function neural network

RBF neural network is a three-layer feed-forward network. The first is the input layer which is composed of signal source nodes. The second is the hidden layer which is composed of radial bases

function in which the number of hidden units is determined by the described problems. The third is the output layer. The inputs of the hidden layer are the linear combinations of scalar weights and the input vector $x = [x_1, x_2, \dots, x_n]^T$ where the scalar weights are usually allocated unity values. The output layer yields a vector $y = [y_1, y_2, \dots, y_n]$ for m outputs by linear combination of the outputs of the hidden nodes to produce the final output (Sun *et al.*, 2009). The network output can be obtained by:

$$y = f(x) = \sum_{i=1}^k w_i \phi_i(x), \tag{10}$$

where $f(x)$ is the final output, $\phi_i(\cdot)$ denotes the radial basis function of the i -th hidden node x is n -dimensional input vector; k is the number of hidden units; w_i is the weight between hidden layer and output layer. The Gaussian activation function is used as basis function in this paper, which is given by:

$$\phi_i = \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right), \tag{11}$$

σ_i is the variance of Gaussian activation function and c is the centre of the Gaussian activation function.

3.3. The proposed PSO-RBF model

It is commonly difficult to fix a certain stochastic model for each inertial sensor which works efficiently in all environments and reflects the long-term behaviour of sensor errors (Noureldin *et al.*, 2004). The fact that KF highly depends on predefined dynamics model forms a major disadvantage (Jwo and Weng, 2008). If the real model is not reflected by the estimated internal model, the KF estimates may not be consistent and a divergence problem may recur. Numerous efforts have been made to ameliorate the covariance matrices estimation. The PSO is employed in this suggested model to obtain relevant RBF parameters. RBF is used to filter the raised frequency noise and to extract the measurement noise from the covariance matrices.

This method consists in training the network on all points except one and tests the error on this point. By repeating this process on all points, we obtain an overall error that is to make minimal.

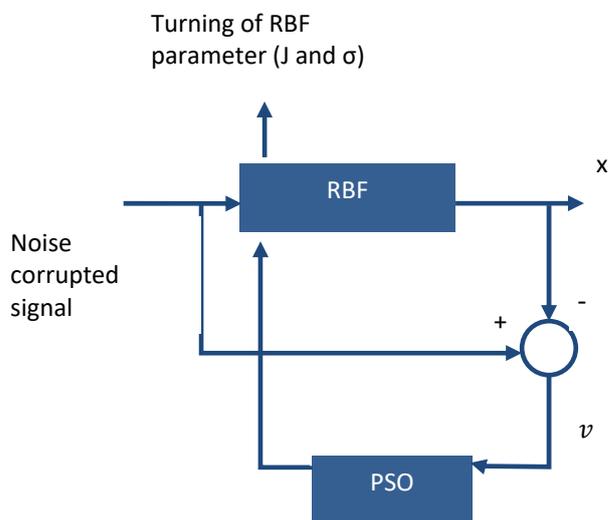


Figure 3. The PSO-RBF functional block diagram

4. Simulation Results

4.1. Simulations

Diverse propagation models (within different sorts of environment) distinguish interactions between the signal and the considered environment. Receiver performance is also taken into consideration in computations.

Here, gives simulation results using our expanded package by MATLAB, we calculate the position and the pseudo-range error using our algorithms. Accordingly, in the simulation, the reception position is well known as well as the pseudo-range error. The number of available satellites varies between 4 and 11 with an average equal to 9.

Adjusting the filters consists in selecting values for these process noises and the measurement noises that optimize the filter performance. A nominal set of adjusting parameters are in Table 1.

Table 1. Setting parameters for the GPSTk

Clock	5 m/sec
Residual troposphere	0.2 m
Phase biases	2 e-6 m/sec
Pseudo-range precision (PR)	5 m
Doppler precision (Dp)	2 Hz

Multipath is not addressed particularly by the algorithm. Pseudo-range multipath is basically detached by the smoothing algorithm presented in (Briers *et al.*, 2010). Phase multipath is minor but in theory a source of position error, though averaging over long times should be adequate to minimize it strongly.

The nominal tropospheric delay is eliminated using any of the numerous classical tropospheric models, as given in the GPSTk (The GPS Toolkit, <http://www.gpskit.org>). The software gets input from RINEX format meteorological files for weather data, or, in their nonexistence, uses default values, which also might be given by the user. The tropospheric delay situation in the KF justifies any residual delay.

The ionospheric delay caused by the ionosphere is displaced to first order in the creation of the ionosphere-free phase. Higher order ionospheric delays may be 10 centimetres large, thus may be a source of error, even though averaging will powerfully lessen their effect. They have not been integrated into the filters, although there are plans to do so.

The C1-P1 bias is a bias linking the C/A and P pseudo-range code on L1, the bias takes part directly to the solution error if C/A code data is used with a similar ephemeris to assess exact positions. In this paper, the pseudo-ranges are only used to debias the phase, consequently any residual bias will take part just a small error of the initial phase bias, and this will certainly be explained in the Kalman filter state.

The ephemerides are recovered from FTP servers of international GNSS Service (IGS Data, <http://sopac.ucsd.edu/dataBrowser.html>) stations and the National Geospatial-Intelligence Agency (NGA, <http://earth-info.nga.mil/GandG/Sathtml/>) respectively, NGA delivers antenna phase centre (APC) satellite orbits and clock data each 15 minutes synchronized to GPS time. All the data are archived in receiver independent exchange (RINEX) n-type format, which contains the ephemeris/Clock parameters broadcast by the satellite, the pseudorandom noise (PRN) and the transmission time of message (TTOM).

The dataset consisting of 10 days (17-26 October of 2014) at each of three different monitor stations. Meteorological information is available for the NGA sites. The NGA (P-code) data was assembled by an Ashtech ZY12 with cesium frequency norm at a 30 second rate.

4.2. Results

The strategy of the PSO-RBF method is the optimization of RBF parameters via the PSO optimization process. To determine the optimal location of the vector particles and the leading particle decoding individual series, the best particles receive the corresponding parameters of the RBF network in every iteration. To prove the performance determining results of the PSO-RBF network, we did simulations under the same condition of PSO, RBF and PSO-RBF network hybrid algorithm then did a comparison.

Measures are the residual of GPS LS data, which are used as the fundament of KF parameter adjustment for the PSO-RBF technique. The experiments of simulation were completed to assess the performance of the proposed method compared to that of conventional methods for GPS navigation processing. Satellite constellation was modelled and the fault origins messed the GPS measurements to encompass tropospheric delay, ionospheric delay, receiver noise and multipath.

Figure 3 shows the PSO-RBF compared to the actual trajectory and the noise corrupted trajectory.

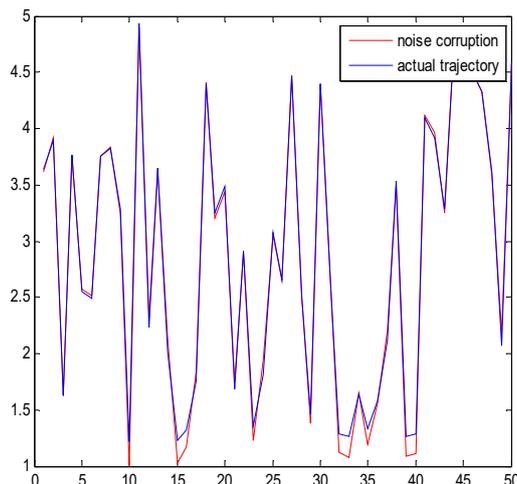


Figure 4. PSO-RBF as compared to the actual trajectory and noise corrupted trajectory

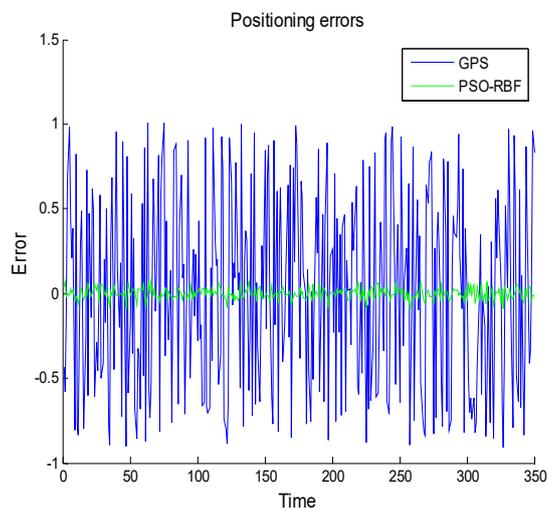


Figure 5. Positioning errors with correction

Figure 4 shows clearly a substantial performance improvement. The fitness function value changes in Figure 5 and it shows that PSO-RBF neural network is more precise. It shows as well that relative time of application PSO-RBF is shorter than the PSO and that there is higher accuracy under the same set of samples. The PSO-RBF algorithm can optimize quickly and can be used for short-term forecasting.

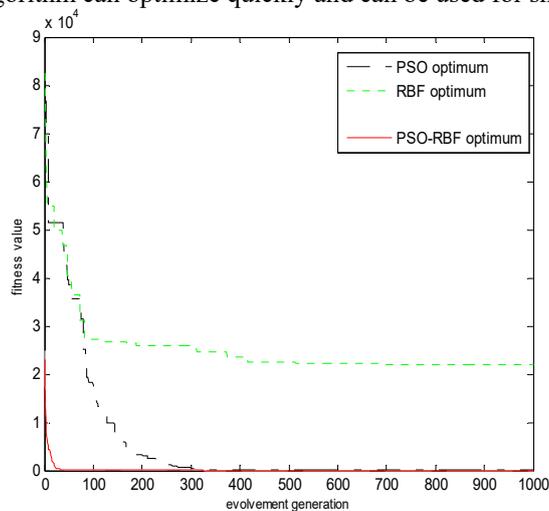


Figure 6. Convergence profile for PSO-RBF

The results of simulation model are in Table 2.

Table 2. Summary table of simulation models

Model	Number of iterations	Precision	Error
RBF	2054	0.001	0.02%
PSO	950	0.001	0.011%
PSO-RBF	590	0.001	0.0029%

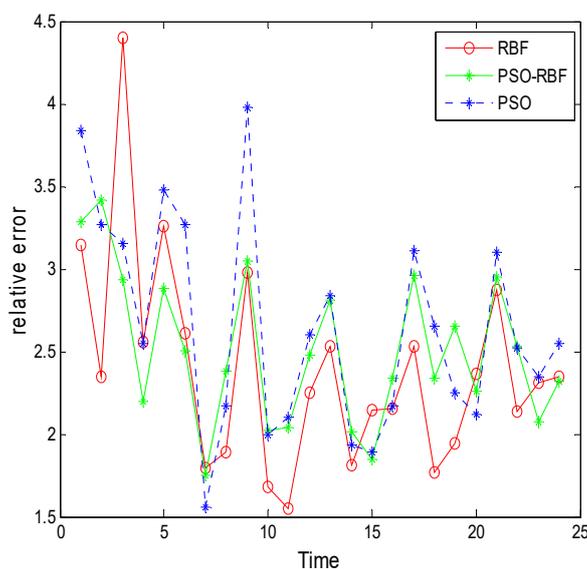


Figure 7. Percentage error of forecasted positioning of a certain day in 24 hours

From Figure 6 we can note that the PSO-RBF neural network error percentage changes uniformly at a rate between 1.5% and 3.4%, consequently the precision of the prediction using the model psoriatic RBF is better than that of the traditional PSO and RBF models.

5. Conclusion

The GPS is one of the primary systems for navigation aids. However, there are many errors associated with the GPS navigation solution. In order to compensate these errors and others (satellite clock error, ephemeris error, multi-pathing, tropospheric delays, ionospheric delays, SA error and etc). In this paper, RBF reaches positive results by solving the nonlinear and non-Gaussian problems. Experimental results reveal that the RBF algorithm is superior to many other particle filtering algorithms. The navigation accuracy based on the proposed PSO-RBF method has been compared to the conventional EKF method and has proved that substantial amelioration in both navigational accuracy and tracking capability. The covariance matrices for both dynamic process and measurement models in the Kalman filter have been estimated on-line and the nonlinear dynamics system with faults have been defined in order to compensate the modelling error. This combination method is based on RBF neural network, and PSO is developed which predicts pseudo-range corrections and compensates the GPS data in real time. The algorithm estimates and maintains the accuracy of the pseudo-range predictions. The results using the proposed approach have proved significant in positioning performance improvement.

References

1. Azami, H., Malekzadeh, M. and Sanei, S. (2013) Optimization of orthogonal polyphase coding waveform for MIMO radar based on evolutionary algorithms, *Journal of mathematics and Computer Science*, Vol. 6, pp. 146-153.
2. Briers, M., Doucet, A. and Maskell, S. (2010) *Smoothing algorithms for state-space models*. *Annals of the Institute of Statistical Mathematics*, Springer, pp.61-89.

3. Cui, X.H. and Polok, T.E. (2005) Document clustering using particle swarm optimization. In: *Proceedings of swarm intelligence symposium, IEEE*, pp. 185–191, Los Alamitos.
4. Dachee Won, *Precise Positioning*, http://smileforday.com/?page_id=84
5. Dehuri, S. and Tripathy, S. (2011) An extended bayesian/HAPSO intelligent method in intrusion detection system. *Knowledge Mining Using Intelligent Agents*, Vol. 6, pp. 133.
6. GPSTk, The GPS Toolkit, <http://www.gpskit.org>.
7. IGS Data, <http://sopac.ucsd.edu/dataBrowser.html>.
8. Jgouta, M. and Nsiri, B. (2015) Statistical estimation of GNSS pseudo-range errors, *Procedia Computer Science*, Elsevier, vol. 73, pp. 258-265.
9. Jwo, D.J. and Weng, T.P. (2008) An adaptive sensor fusion method with applications in integrated navigation, *Journal of Navigation*, Vol. 61, No. 4, pp.705-721.
10. Leonard, J.A. and Kramer, M.A. (1991) Radial basis function networks for classifying process faults, *Control Systems, IEEE*, Vol. 11, No. 3, pp. 31–38.
11. Li, J. and Li, B. (2014) Parameters selection for support vector machine based on particle swarm optimization. In *Intelligent Computing Theory*, Springer International Publishing, pp. 41-47.
12. Mosavi, MR. and Rahemi, N. (2015) Positioning performance analysis using RWLS algorithm based on variance estimation methods, *Aerospace Science and Technology*, pp.88-96.
13. Mussi, L., Cagnoni, S. and Daolio, F. (2009) GPU-based road sign detection using particle swarm optimization, *International Conference on Intelligent Systems Design and applications*, pp. 152-157.
14. NGA, National Geospatial-Intelligence Agency, <http://earth-info.nga.mil/GandG/Sathtml/>.
15. Noureldin, A., Osman, A. and Elsheimy, N. (2004) A neuro-wavelet method for multi-sensor system integration for vehicular navigation, *Measurement Science and Technology*, Vol. 15, No. 2, pp. 404-412.
16. Santerre, R. Roy, E. and Parrot, D. (1995) Positionnement GPS avec des mesures de pseudo distance filtrées et lissées, *Lighthouse-Burlington*, pp. 21-30.
17. Shen, C. Cao, G.Y and Zhu, X.J. (2002) Nonlinear modelling of MCFC stack based on RBF neural networks identification, *Simulation Modelling Practice and Theory*, Vol.10, No.1, pp.109-119.
18. Sun, T-Y., Liu, C-C., Lin, C-L., Hsieh, S-T. and Huang, C-S. (2009) *A radial basis function neural network with adaptive structure via particle swarm optimization*. - http://www.intechopen.com/books/particle_swarm_optimization/a_radial_basis_function_neural_network_with_adaptive_structure_via_particle_swarm_optimization.