An Advanced GPS Carrier Tracking Loop Based on Neural Networks Algorithm

Jichun Shen, Shuai Chen, Changhui Jiang, Yuming Bo

Abstract— **The GPS (Global Positioning Systems) is the main tool to provide the PVT (position, velocity and time) service in our daily life. However, there are some drawbacks needed to be overcome for the wider applications of GPS. It is hard for a popular GPS receiver to position in a high dynamic/weak signal environment without outer aiding. On the basis of analyzing the reason and the error sources of the tracking loop, a neural networks algorithm is used to adjust the parameters of the fusion of the 2rd FLL and 3rd PLL. A judge factor is selected to present the phase errors of the carrier loop. A nonlinear function between a judge factor and fusion parameters is constructed by the neural network algorithm. The method essentially changes the bandwidth of the tracking loop by changing the loop gain, which is usually ignored. The algorithm is implemented and tested in a Matlab software receiver. The experiments show the modified tracking carrier loop can work well in higher dynamic environments compared with standard carrier tracking loop.**

Index Terms— **GPS, carrier tracking, neural networks, high dynamic.**

I. Introduction

Since the Global Positioning System developed by the United States Department of Defense, the GPS has been widely used as a low-cost equipment to provide precise position information and velocity information and time. GPS is a satellite-based all-weather navigation system. For the Global Positioning System, the satellites broadcast navigation signal to all the earth. And the users' receiver get the signal from the satellites [1]. Therefore, only the receiver is in the direct line to the satellite, the receiver can get the correct signals which can be applied to navigation. However, the system cannot provide continues navigation information in high dynamic environment. The traditional carrier tacking loop cannot tolerate the dynamic stress.

For the commonly used GPS receivers, the carrier tracking loops are composed of frequency locked loop (FLL) and phase locked loop (PLL). The FLL discriminators produce the frequency error of the incoming signal and replica signal. The PLL discriminators produce the phase error of the incoming signal and the replica signal. The FLL usually performs better than PLL in high dynamic environment. Fig1 is the block diagram of a general GPS receiver carrier tracking loop. Carrier pre-detection integration, the loop discriminators and the loop filter are the key three components of the carrier tracking loop [1]. The discriminator outputs are the frequency/phase errors of the incoming signal and the local replica signal, which is the inputs of the loop filter. They are the most important factors which affect the thermal noise and the LOS (line of sight) dynamic stress threshold. In fact, the thermal noise and the dynamics stress are a paradox, the narrow loop filter bandwidth brings reduced thermal noise and the dynamic stress needs wide bandwidth [2]. In practice, a compromise must be made to design a receiver or give up the loop filter with a brand new filter.

In recent decades, with the rapid development of modern computer technology in the software and hardware, artificial neural networks (ANNs) have been applied to a wide variety of complicated problems [3]. It origins from the research of the principles of the human brain neuron network. The ANN is one of the artificial intelligence algorithms which has the ability to learn and understand. It is the abstraction from the information processing of human brain neuron network, and establish some simple model, according to the different composition of different network connections. Neural network is a computing model, by a large number of nodes (or neurons). Each node represents a specific output function which is called activation function (activation function). Each connection between two nodes represents a weighted value for the connection signal, known as a weight, which is equivalent memory artificial neural network. The output of the network depends on the network connection, and the different weights and different excitation functions.

Due to the ability to solve complex problems, the algorithm has been used to solve problems in guidance and navigation and control. Dah-Jing Jwo [4] applied back-propagation neural networks to GDOP approximation and explored the performance and computational benefit of neural network-based GDOP approximation. Kai-Wei Chiang [5] developed an intelligent navigator to provide reliable navigation information during the GPS outages. The research strongly indicate the potential of the intelligent navigator. Naser EI-Sheimy [6] utilize neural network for multisensor system integration in navigation and positioning instruments. In all previous cases, the neural networks has never been used in carrier tracking loops. In this paper, a back propagation neural network is incorporated with traditional FLL-assisted-PLL carrier tracking loop to solve the paradox of thermal noise and dynamic stress. A nonlinear functions of PLL phase error judge indicators and the parameters of the loop filter inputs.

Jichun Shen, School of Automation, Nanjing University of Science and Technology, Nanjing, China, (+86)18248781511.

Shuai Chen, School of Automation, Nanjing University of Science and Technology, Nanjing, China, (+86)13813915826.

Changhui Jiang, School of Automation, Nanjing University of Science and Technology, Nanjing, China, (+86)13770916637.

Yuming Bo, School of Automation, Nanjing University of Science and Technology, Nanjing, China.

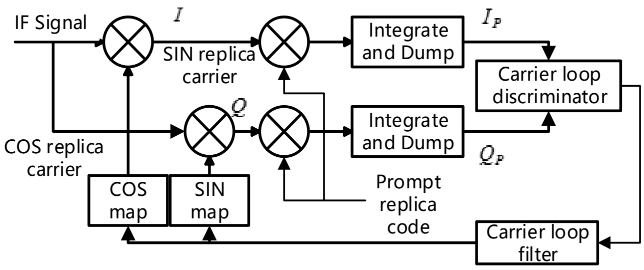

Fig. 1. Structure of generic GPS receiver carrier tracking loop

The rest of the paper is organized as follows. The second section of the paper is the introduction of the system design and details of the applied back-propagation artificial neural networks. The next is the performance evaluation of the algorithm and the conclusions.

II. Design of the Proposed Carrier Tracking Loop

A. *Structure of the Tracking Loop*

Generally, the traditional tracking loops are composed of frequency locked loop (FLL) and phase locked loop (PLL). The FLL discriminators produce the frequency error of the incoming signal and replica signal. The PLL discriminators produce the phase error of the incoming signal and the replica signal. When the carrier tracking loop locks well, the phase error is near to zero. The value of the phase error reflects the quality of the carrier tracking loop. An indicator factor is selected to judge the quality of the tracking loop.

$$E = \cos(2\Delta\varphi) = \frac{I_P^2 - Q_P^2}{I_P^2 + Q_P^2}$$

According to the characteristics of FLL and PLL, the PLL is sensitive to the dynamic stress, but it can provide more accurate velocity measurements compared with FLL. The FLL performs better in high dynamic environment. In order to design a robust receiver carrier tracking loop, the FLL should operate at wide bandwidth and the PLL operate at narrow bandwidth.

A Back-propagation neural networks is applied to construct the nonlinear functions of the indicator factor and the fusion parameters. The fusion parameters are adjusted according to the values of the indicator factor, which makes the carrier tacking loop to tolerate the thermal noise and the dynamic stress. The Fig2 detailed the design and the structure of the system. The back-propagation artificial networks algorithm was employed in the design.

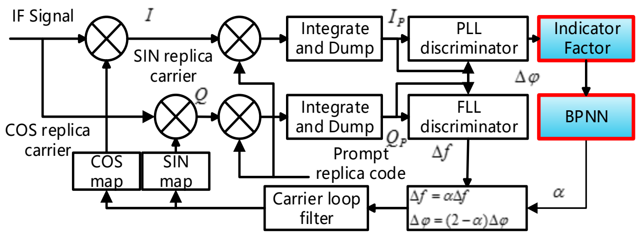

Fig. 2. Scheme of the advanced carrier tracking loop.

B. *The Back-propagation Neural Networks*

Neural networks (NN) are trainable and have the ability to learn and understand, which can be used to map the input-output functions. They have been applied to solve a wide variety of complicated problems which cannot be processed with conventional theory. Because the NNs can be used as a 'Black Box' and don't need to describe the mathematics model. The most important is the way how a neuron is implemented and how they are connected. The back-propagation is the one of the most practical learning algorithms which learn and understand input-output functions from selected samples. The BPNN is a supervised algorithm which calculated the values of the gradient in the opposite direction of the flow of each node output [7]. When you submit your final version, after your paper has been accepted, prepare it in two-column format, including figures and tables.

BP neural network consists of an input layer and an output layer and one or more hidden layers, which are independent of each other. Input signals from the input neurons in turn through the various hidden layer neurons, the final output to the output neurons (Fig3). The input is the values of the indicator factor and the output is the fusion parameter of the loop filter (Fig2). The essence of BP algorithm is to find the minimum value of the error function. In this algorithm, the weighted coefficient of the negative gradient direction of the error function is modified by the steepest descent method in nonlinear programming.

BP uses a function called "active function" to describe the relationship between the output of the different layers, thus simulating the interaction between the layers. So the more commonly used active function is called S type [7]:

$$f(x) = \frac{1}{(1+\exp(-x))}$$

This function is smooth and continuous and differentiable.

$$\dot{f}(x) = x(1-x)$$

Where: $x \in (-\infty, +\infty)$, and $f(x) \in (0,1)$.

The connection weights of the network are changed constantly under the stimulation of the external input sample, which is the dynamic adjustment of the weights of the connections. The core is the weight adjustment rules, which is the adjustments rules in the process of changing the connection weights of each neuron of the layers.

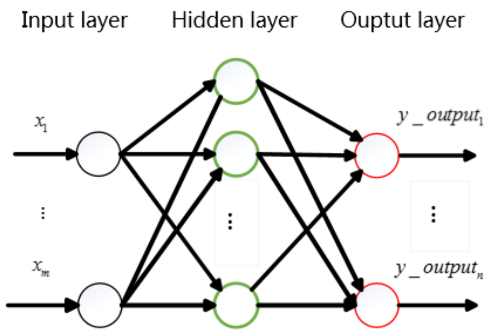

Fig. 3. A three-layer of back-propagation neural network

The nonlinear function of the two variables is built by training the back-propagation neural networks based on the input samples and the output samples. And then it can be used as a "Black Box" in the design. The training procedures can be explained as following steps [7]:

Step One: Set the initial values of w_{ij}, w_{jk}, b_j, b_k.

Step Two: Input the training input samples and the desired outputs.

$$x\_input = (x_1, x_2, x_3, ..., x_m)$$

$$d\_ouput = (d_1, d_2, d_3, ..., d_n)$$

Step Three: Calculate the inputs and outputs for the j^{th} hidden neuron:

$$h\_input_j = \sum_{i=1}^{m} w_{ij} x_i - hidden\_b_j$$

$$h\_output_j = f(h\_input_j)$$

Step Four: Calculate the inputs and outputs for the k^{th} output neuron:

$$y\_input_k = \sum_{j=1}^{p} w_{jk} h\_output_j - ouput\_b_k$$

$$y\_output_k = f(y\_input_k)$$

Step Five: Calculate the error of the k^{th} output neuron.

$$\delta_k = (d_k - y\_output_k)\dot{f}(y\_input_k) =$$
$$y\_output_k(1 - y\_output_k)(d_k - y\_output_k)$$

Step Six: Calculate the error of the j^{th} hidden neuron.

$$\delta_j = \dot{f}(h\_input_j)\sum_{k=1}^{n} w_{jk}\delta_k =$$

$$h\_input_j(1 - h\_input_j)\sum_{k=1}^{n} w_{jk}\delta_k$$

Step Seven: Correct the connect weights and thresholds between input layer, hidden layer and output layer.

$$\Delta w_{jk} = \eta\delta_k h\_output_j \quad \Delta output\_b_k = \eta\delta_k$$

$$\Delta w_{ij} = \eta\delta_j x_i \quad \Delta hidden\_b_j = \eta\delta_j$$

Update w_{jk} and $output\_b_k$:

$$w_{jk} = w_{jk} + \Delta w_{jk},$$

$$output\_b_k = output\_b_k + \Delta output\_b_k$$

Update w_{ij} and $output\_b_j$:

$$w_{ij} = w_{ij} + \Delta w_{ij}, \; output\_b_j = hidden\_b_j + \Delta hidden\_b_j$$

Calculate the error cost function:

$$error\_\cos t = \frac{1}{2}\sum_{i=1}^{n}(y\_output_i - d\_output_i)^2$$

Repeat the steps until the cost function reaches to a desired range.

Where m is the number of the input layer neurons and p is the number of the hidden layer neurons and n is the number of the output layer neurons.

Input samples vector:

$$x\_input = (x_1, x_2, x_3, ..., x_m)$$

Desired output samples vector:

$$d\_output = (d_1, d_2, d_3, ..., d_n)$$

Input vector of hidden layer:

$$h\_input = (h\_input_i, h\_input_2, ..., h\_input_p)$$

Output vector of hidden layer:

$$h\_output = (h\_output_i, h\_output_2, ..., h\_output_p)$$

Input vector of output layer:

$$y\_input = (y\_input_1, y\_input_2, ..., y\_input_n)$$

Output vector of output layer:

$$y\_output = (y\_output_1, y\_output_2, ..., y\_output_n)$$

Learning rate: η

The connection weights of the input layer neurons and the output layer neurons: w_{ij}

The connection weights of the hidden layer neurons and the output layer neurons: w_{jk}

Thresholds of the hidden layer neuron: b_j

Thresholds of the output layer neuron: b_k

Error function:

$$error\_\cos t = \frac{1}{2}\sum_{i=1}^{n}(y\_output_i - d\_output_i)^2$$

III. TEST AND SIMULATION

Finally, a test platform and tool is necessary for the evaluation of the algorithm. A software designed receiver is a low-cost and flexible tool to operate and evaluate the performance of the proposed scheme.

A. GPS IF Signal Collector

A hardware GPS signal simulator is necessary for the test of the proposed algorithm. Because it is the hard to verify the algorithm in real high dynamic environment. The algorithm is implemented and tested in a Matlab GNSS software receiver developed by our lab research group. Fig5 shows the details of GPS IF data collecting equipment. Several high dynamic scenarios are designed and simulated in the hardware signal simulator. The GPS IF data are collected by the IF signal collector after the signal passing through the antenna due to the computer cannot directly process the GPS high-frequency signal data. The sampling frequency of the collected GPS L1 signal is 16.369MHz and the intermediate frequency is 3.996MHz, the data type is ‘int’.

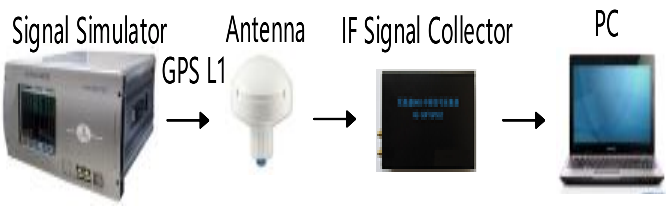

Fig. 4. The flow chart of the collected GPS L1 signal

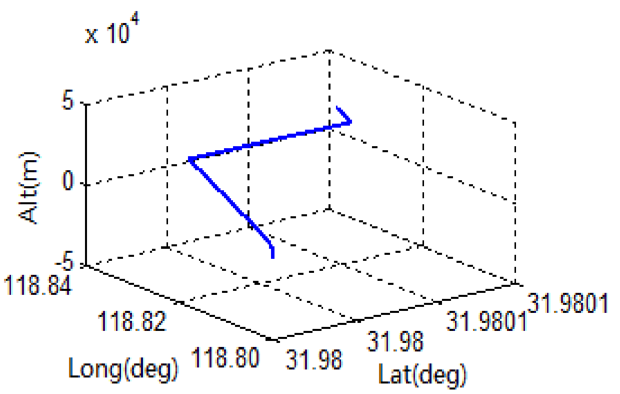

Fig. 5. 3D plot of the selected trajectory

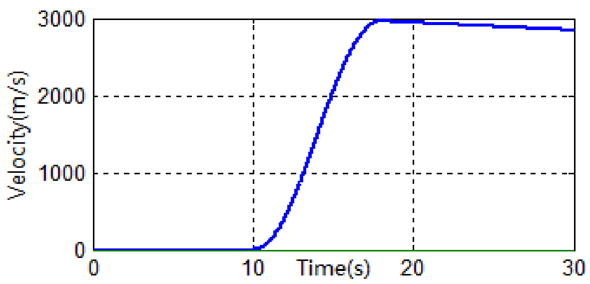

Fig. 6. Velocity

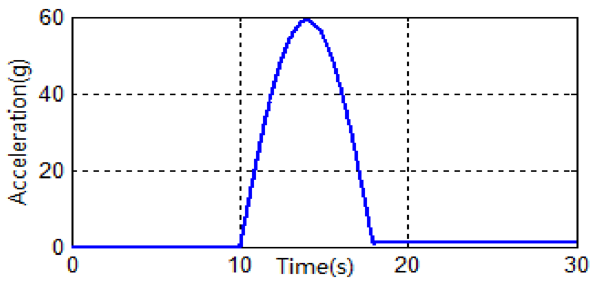

Fig. 7. Acceleration

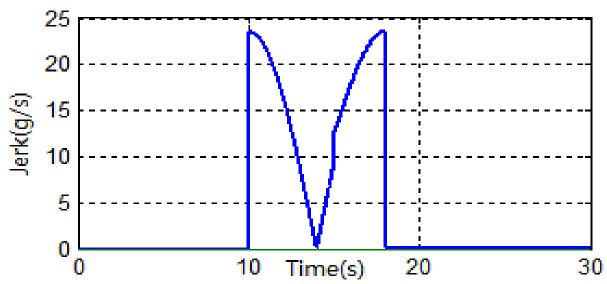

Fig. 8. Jerk

IV. RESULTS

Two different tracking schemes are employed for GNSS signals tracking of the same high dynamic trajectory. They include FLL-assisted-PLL (40Hz, 25Hz) and the FLL-assisted-PLL based on BPNN (FBPPLL). Tracking performance of the two schemes are evaluated according to the phase error of the phase discriminator and the I and Q values in the same high dynamic trajectory.

A. Performance Evaluation

The main purpose of the carrier tracking loop is to tracking the navigation signal and replica it. The phase errors is the output of the PLL discriminator which can be a practical index to measuring the tracking performance. Fig9, Fig10, Fig11 and Fig12 shows the calculated phase errors of four different tracking channels for the same high dynamic trajectory. The FBPPLL phase error is much smaller than the conventional tracking algorithm (40Hz FLL and 25Hz PLL PIT 1ms). Fig13, Fig14, Fig15and Fig16 shows the I and Q values of the carrier tracking loop. When the tracking loop works well. The I is composed of the navigation data and the Q is just noise. Fig8 shows the Q values of the FBPPLL is much smaller than the conventional carrier tracking loops.

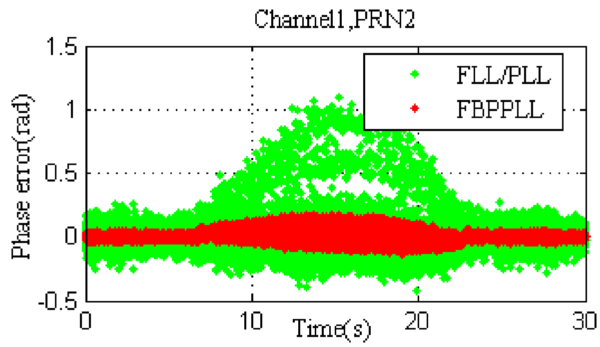

Fig. 9. Output of the PLL discriminator

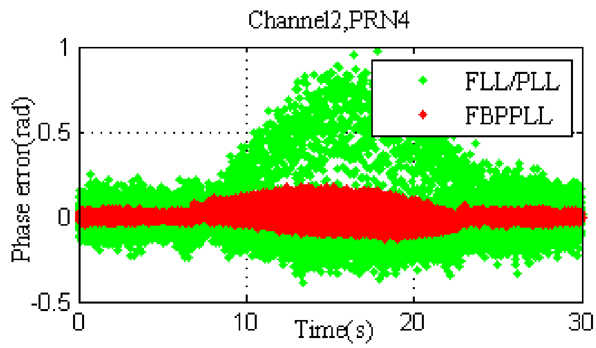

Fig. 10. Output of the PLL discriminator

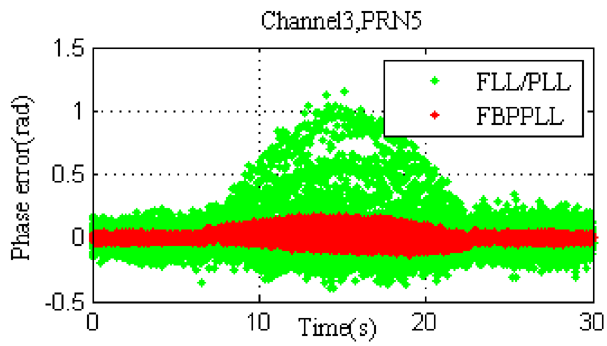

Fig. 11. Output of the PLL discriminator

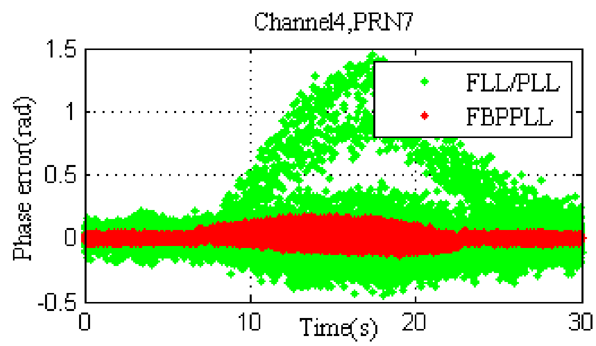

Fig. 12. Output of the PLL discriminator

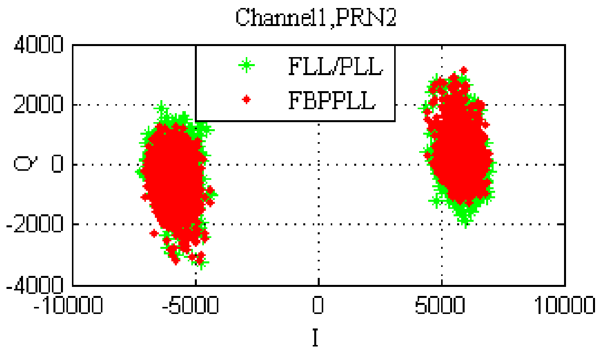

Fig. 13. Output of the corrector

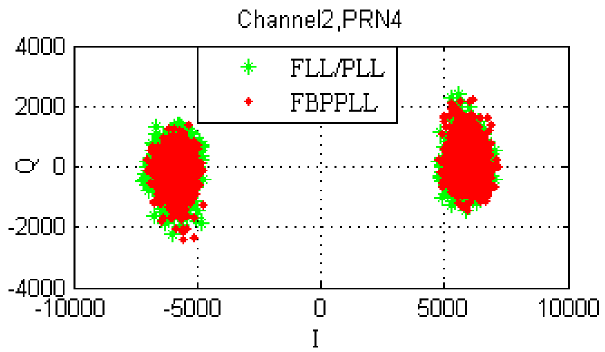

Fig. 14. Output of the corrector

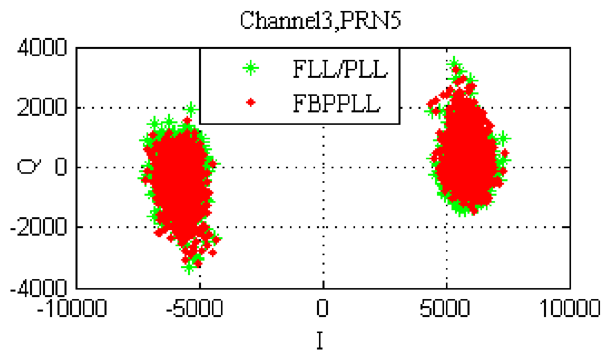

Fig. 15. Output of the corrector

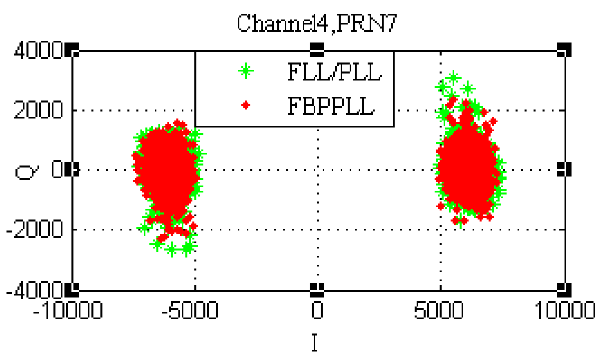

Fig. 16. Output of the corrector

V. Conclusion

In this paper the BPNN algorithm has been applied to the carrier tracking loop which is unique and distinct from any other carrier tracking loops. A nonlinear function has been constructed between the indicator factor and the fusion parameters. The method solves the bandwidth paradox of the thermal noise and dynamic stress and make the carrier loop work well in high dynamic.

Acknowledgment

This work was supported by National Natural Science Foundation of China (Grant No.U1330133), the Fundamental Research Funds for the Central Universities (No.30916011336), Jiangsu Planned Projects for Postdoctoral Research Funds (1501050B), and the China Postdoctoral Science Foundation (2015M580434). This work also got the special grade of the financial support from the China Postdoctoral Science Foundation (2016T90461).

References

[1] Kaplan, E.D (2006) Understanding GPS: Principles and applications, Second Ed., Artech House, INC., Norwood, MA 02062

[2] Ward, P. (1998) “Performance Comparisions Between FLL, PLL and a Novel FLL-Assisted-PLL Carrier Tracking Loop Under RF Interference Conditions.” Proceedings of the 11th International Techinal Meeting of The Satellite Division of The Institude of Navigation, September 1998, Nashville, TN,pp. 783-795

[3] Poggie, T. and Girosi, F. Networks for approximation and learning. Proc. of the IEEE 78(9), pp1481-1497, 1990.

[4] Dah-Jing Jwo. “Applying Back-propagation Neural Networks to GDOP Approximation.” THE JOURNAL ONAVIGATION, vol.55, pp. 97-108. 2002.

[5] Kai-Wei Chiang. “An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications.” Applied Computing, vol.8, pp 722-733, 2007.

[6] Naser El-Sheimy. “The Utilization of Artificial Neural Networks for Multisensor System Integration in Navigation and Positioning Instruments.” IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUEMNT, vol. 55 (no.5), 2005.

[7] Haykin, S. Neural Networks: A Comprehensive Foundation. Prentice-Hall, 1990.

Jichun Shen is now pursuing his bachelor degree in Control Science and Engineering from Nanjing University of Science and Technology. His research now focuses on GNSS/SINS deeply coupled navigation system.

Changhui Jiang is now pursuing his Ph. D in Control Science and Engineering from Nanjing University of Science and Technology. He received his B.E from Soochow University in 2014. His research now focuses on GNSS/SINS deeply coupled navigation system.

Yuming Bo received his M.Sc. degree in Control Science and Engineering at Nanjing University of Science and Technology, China, in 1987. He received his Ph.D. degree in Control Science and Engineering at Nanjing University of Science and Technology, China, in 2005. He is currently a Professor in School of Automation at Nanjing University of Science and Technology, China. His research interests include information processing, navigation, guidance, control theory and application.