Journal of Geographic Information System, **2011, 3, 318-322**
doi:10.4236/jgis.2011.34029 Published Online October 2011 (http://www.SciRP.org/journal/jgis)

Performance Improvement of GPS GDOP Approximation Using Recurrent Wavelet Neural Network

Sadaf Tafazoli, Mohammad Reza Mosavi*
Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran
E-mail: **M_Mosavi@iust.ac.ir*

Received May 5, 2011; *revised June* 26, 2011; *accepted July* 14, 2011

Abstract

One of the most important factors affecting the precision of the performance of a GPS receiver is the relative positioning of satellites to each other. Therefore, it is essential to choose appropriate accessible satellites utilized in the calculation of GPS positions. Optimal subsets of satellites are determined using the least value of their Geometric Dilution of Precision (GDOP). The most correct method of calculating GPS GDOP uses inverse matrix for all combinations and selecting the lowest ones. However, the inverse matrix method, especially when there are so many satellites, imposes a huge calculation load on the processor of the GPS navigator. In this paper, the rapid and precise calculation of GPS GDOP based on Recurrent Wavelet Neural Network (RWNN) has been introduced for selecting an optimal subset of satellites. The method of NNs provides a realistic calculation approach to determine GPS GDOP without any need to calculate inverse matrix.

Keywords: Rapid and Precise Calculation, GPS GDOP, RWNN

1. Introduction

Global Positioning System (GPS) is a satellite based positioning system which was rapidly grown in the past two decades. It can cover almost all around the world, using satellite signals in order to measure accurate time, altitude, longitude and latitude in every desirable point on earth, sea or air as well as space.

The methods of calculating the coordinates in a GPS receiver are based on the utilizing four visible satellites in a set of visible satellites and errors may usually happen when one or more satellites are invisible or the information sent by them are unclear. There are various algorithms for the classification of visible satellites that can be used to classify appropriate satellites in one group. Considering the geostationary position of satellites in earth's orbit, five to eight GPS satellites are visible at any position on earth. To calculate the coordinates of that point on the earth the four-satellite groups can be used differently [1,2].

In the common algorithm used for the most GPS receivers, the first selected satellite is the one whose connecting line joining the satellite to its receiver is more vertical. After selecting the base satellite, the other three satellites are determined based on their most appropriate geometric configuration. At the beginning of locating and after appropriate selection of four satellites according to the said algorithm, the error occurring within a specified limit is acceptable; but the increase in the time of the original selection increases proportionally the occurred error. Therefore, it is required that the selection is repeated within specified intervals. To reduce the errors of calculation, it is necessary to reduce the temporal interval existing between the classification and frequent selections of appropriate satellites. The increase rate of calculation error after selection, and fixation of satellites depends on the different parameters including manufacturing technology of receiver, quality of receiver, applied algorithms, and number of tracking channels. Any increase in tracking channels of a receiver increases the ability of the receiver in selecting and classifying satellites. Therefore, the receivers having more tracking channels cause less error than other receivers. As the vehicles using the GPS moves, the initial selected satellites disappear in horizon and become invisible to the human eye. At this stage, other appropriate satellites shall be selected [3,4].

The best method for calculating the Geometric Dilution of Precision (GDOP) of GPS satellites is to use inverse matrix for all configurations and selecting the smallest one; but, the inversion of matrix imposes a huge calculation load on the processor of the navigator [5].

The use of Neural Networks (NNs) is a solution for complicated issues, for which no mathematical models are available. NNs use several simple computing units called neuron (nervous cells) patterned after the cells of human brain. The neurons of each NN process and convert the stimulations or input data to send them to the outputs. These outputs may be connected to the inputs of other neurons. These neurons connecting to each other form a NN. A NN consists of one input and one output layer. The data are received by a NN through its input layer and processes by all layers before receiving the output layer [6].

The objective of this paper is to suggest a rapid method for calculating GPS GDOP using the NNs. This method strongly decreases the load of GPS GDOP classical method calculations for selecting an optimal subset of satellites by GPS navigator processors. This paper has been organized as follows. In the Section 2, the concept of GPS GDOP has been studied in brief. In the Sections 3, 4 and 5, the manner of rapid and precise calculation of GPS GDOP based on RWNN to select an appropriate subset of navigator satellites. To study the workability of this method, it has been tested and compared in the Section 6. Finally, the Section 7 provides us with the conclusion.

2. The Concept of GPS GDOP

GPS receivers report usually the geometric quality of satellites based on Position Dilution of Precision (PDOP). PDOP shows the vertical and horizontal DOP, *i.e.* geographical longitude, latitude, and altitude. Low DOP increases and high DOP decreases the probability of precision. If the sides of the pyramid formed by four satellites are almost equal (equiangular pyramid), this configuration leads to an appropriate GPS GDOP and vice versa. GPS GDOP is a very effective tool for GPS. All receivers use the algorithms based on GPS GDOP to find the best subset of visible satellites used for tracking. To locate satellites, azimuth (AZ) and angle of elevation (E) can be used. To define the concept of GPS GDOP, it is helpful to use sample calculations expressing a compromise between precision and the position of satellite. For this purpose, the matrix of H is defined as follows [7]:

$$H = \begin{bmatrix} \text{Cos}(E1)*\text{Sin}(Az1) & \text{Cos}(E1)*\text{Cos}(Az1) & \text{Sin}(E1) & 1 \\ \text{Cos}(E2)*\text{Sin}(Az2) & \text{Cos}(E2)*\text{Cos}(Az2) & \text{Sin}(E2) & 1 \\ \text{Cos}(E3)*\text{Sin}(Az3) & \text{Cos}(E3)*\text{Cos}(Az3) & \text{Sin}(E3) & 1 \\ \text{Cos}(E4)*\text{Sin}(Az4) & \text{Cos}(E4)*\text{Cos}(Az4) & \text{Sin}(E4) & 1 \end{bmatrix} \tag{1}$$

GPS GDOP shall be defined as follows:

$$\text{GDOP} = \sqrt{\text{trace}(H^T H)^{-1}} = \sqrt{\frac{\text{trace}[\text{adj}(H^T H)]}{\det(H^T H)}} \tag{2}$$

GPS GDOP provides us with a simple interpretation of the fact that how much a unit of measurement error participates in the occurrence of positioning error for a specified position, and it determines the factor of measurement noise amplitude.

3. Rapid and Precise Calculation of GPS GDOP Using Neural Networks

To prepare experimenting data, all input and output variables are normalized within an interval of [0,1] to reduce the experimenting time. whereas, $H^T H$ is a 4 × 4 matrix, it has four eigenvalues λ_i ($i = 1, 2, 3, 4$).

We know that four eigenvalues for the matrix of $\left(H^T H\right)^{-1}$ is equal to λ_i^{-1}. Considering that the trace of a matrix is equal to the sum of its eigenvalues, the following equation is formed as follows [8]:

$$\text{GDOP} = \sqrt{\lambda_1^{-1} + \lambda_2^{-1} + \lambda_3^{-1} + \lambda_4^{-1}} \tag{3}$$

The mapping with the definition of four variables is as follows:

$$x_1 = \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = \text{trace}(H^T H) \tag{4}$$

$$x2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2 + \lambda_4^2 = \text{trace}\left[(H^T H)^2\right] \tag{5}$$

$$x_3 = \lambda_1^3 + \lambda_2^3 + \lambda_3^3 + \lambda_4^3 = \text{trace}\left[(H^T H)^3\right] \tag{6}$$

$$x_4 = \lambda_1 \lambda_2 \lambda_3 \lambda_4 = \det(H^T H) \tag{7}$$

GPS GDOP can be shown as a functional mapping of $R^4 \to R^1$ from $\vec{f}$ directly to GPS GDOP; *i.e.* $\vec{\lambda}^{-1} = \text{fn}\left(\vec{f}\right)$:

Input: $\left(x_1, x_2, x_3, x_4\right)^T$

Output: $y = \text{GPS GDOP}$

The mapping from $\vec{f}$ to GPS GDOP is strictly non-linear and cannot be determined analytically, but it can be approximated using a NN. The NN used in this paper has been designed for the mapping of $R^4 \to R^1$ from $\vec{f}$ to GPS GDOP. The **Figure 1** shows the overall diagram block of GPS GDOP approximation using NNs including Recurrent NN (RNN), Wavelet NN (WNN), and Recurrent Wavelet NN (RWNN).

4. RWNN Architecture

The RWNN topology is made of tree layers, one input layer, one hidden layer, and one output layer. **Figure 2** presents a kind of tree layer RWNN structure. The input layer has M nodes. The output layer also has only one neuron whose output is the signal represented by the

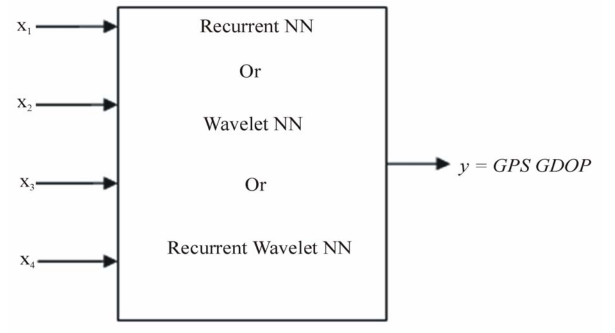

Figure 1. The overall diagram block of GPS GDOP approximation using NNs.

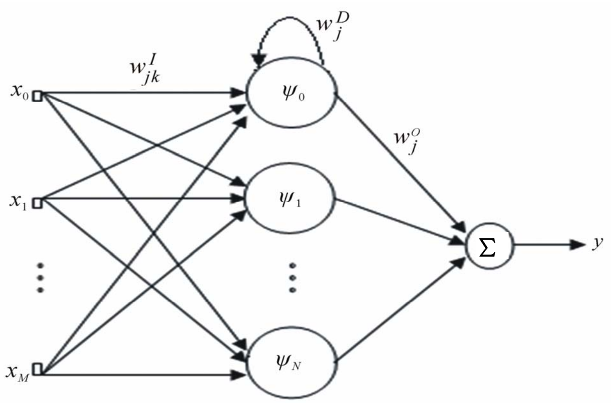

Figure 2. RWNN architecture with (*M* + 1, *N* + 1, 1) structure.

weighted sum of several wavelets. The hidden layer is composed of finite number of wavelets representing the signal.

The components of the proposed RWNN essentially include: x_k -value of the k -th input neuron; ψ_j -output of the j -th hidden neuron; w_{jk}^I -interconnection weight between the k -th input neuron and j -th hidden neuron; w_j^O -interconnection weight between the j -th hidden neuron and the output neuron; w_j^D -recurrent weight for j -th hidden neuron.

The net internal activity of neuron j at time *n*, is given by:

$$net_j(n) = \sum_{k=0}^{M} w_{jk}^I(n)x_k(n) + w_j^D \psi_{a,b}[net_j(n-1)] \quad (8)$$

where, $net_j(n)$ is the sum of input to the j -th recurrent neuron, $\psi_{a,b}\left[net_j(n)\right]$ is the output of the j -th recurrent neuron and is computed by passing $net_j(n)$ through the wavelet function $\psi_{a,b\,j}(.)$, obtaining:

$$\psi_{a,b}\left[net_j(n)\right] = \psi\left[\frac{net_j(n) - b_j(n)}{a_j(n)}\right] \quad (9)$$

where, $a_j(n)$ and $b_j(n)$ are the dilation and translation coefficients of the j -th wavlon in hidden layer, respectively. The RWNN output of the **Figure 2** network is computed:

$$y(n) = \sum_{J=0}^{N} w_j^O(n)\psi_{a,b}\left[net_j(n)\right] \quad (10)$$

The wavelet function which we have considered here is the called "Gaussian-derivative" function as:

$$\psi(x) = -xe^{-\frac{1}{2}x^2} \quad (11)$$

5. Learning Algorithm for RWNN

The basic principle of the RWNN is to use the gradient steepest descent method to minimize the cost function. The training of RWNN is traditionally based on minimization of the cost function. Suppose a set of training samples is available, the problem can be characterized as choosing the weights (or coupling strengths) of a given network such that the following total squared error is minimized [9]:

$$E(n) = \frac{1}{2}e^2(n) = \frac{1}{2}\left[d(n) - y(n)\right]^2 \quad (12)$$

where $d(n)$ and $y(n)$ represent the desired and actual output of the output neuron, respectively. $e(n)$ is a time varying error. The output weights can be adjusted according to:

$$\begin{aligned} w_j^O(n+1) &= w_j^O(n) + \eta\frac{\partial E(n)}{\partial w_j^O(n)} \\ &= w_j^O(n) + \eta e(n)\frac{\partial y(n)}{\partial w_j^O(n)} = w_j^O + \eta e(n)\psi_{a,b}\left[net_j(n)\right] \end{aligned} \quad (13)$$

where, η is a learning rate. The recurrent weights are updated as follow:

$$w_j^D(n+1) = w_j^D(n) + \eta\frac{\partial E(n)}{\partial w_j^D(n)} = w_j^D(n) + \eta e(n)\frac{\partial y(n)}{\partial w_j^D(n)} \quad (14)$$

To determine the partial $\frac{\partial y(n)}{\partial w_j^D(n)}$ derivative, is differentiated the network dynamics with respect to $w_j^D(n)$ as follow:

$$\frac{\partial y(n)}{\partial w_j^D} = w_j^O(n)\frac{\partial \psi_{a,b}[net_j(n)]}{\partial w_j^D(n)} \quad (15)$$

where $\frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial w_j^D(n)}$ is done by using the chain rule for differentiation, obtaining:

$$\begin{aligned}\frac{\partial \psi_{a,b}[net_j(n)]}{\partial w_j^D(n)} &= \frac{\partial \psi_{a,b}[net_j(n)]}{\partial net_j(n)}\frac{\partial net_j(n)}{\partial w_j^D(n)} \\ &= \psi'_{a,b}\left[net_j(n)\right]\frac{\partial net_j(n)}{\partial w_j^D(n)}\end{aligned} \tag{16}$$

Or:

$$\begin{aligned}\frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial w_j^D(n)} &= \frac{\psi'_{a,b}\left[net_j(n)\right]}{a_j(n)}\times \\ &\left[\psi_{a,b}\left[net_j(n-1)\right] + w_j^D(n)\frac{\partial \psi_{a,b}\left[net_j(n-1)\right]}{\partial w_j^D(n)}\right]\end{aligned} \tag{17}$$

Equation (17) is non-linear dynamic recursive equation and can be solved recursively with given initial condition as:

$$\frac{\partial \psi_{a,b}(0)}{\partial w_j^D(0)} = 0 \tag{18}$$

The inputs weight can be adjusted as follow:

$$\begin{aligned}w_{jk}^I(n+1) &= w_{jk}^I(n) + \eta\frac{\partial E(n)}{\partial w_{jk}^I(n)} \\ &= w_{jk}^I(n) + \eta e(n)\frac{\partial y(n)}{\partial w_{jk}^I(n)} \\ &= w_{jk}^I(n) + \eta e(n) w_j^O(n)\frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial w_{jk}^I(n)}\end{aligned} \tag{19}$$

where $\frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial w_{jk}^I(n)}$ is obtained as:

$$\begin{aligned}\frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial w_{jk}^I(n)} &= \frac{\partial \psi_{a,b}\left[net_j(n)\right]}{\partial net_j(n)}\frac{\partial net_j(n)}{\partial w_{jk}^I(n)} \\ &= \psi'_{a,b}\left[net_j(n)\right]\frac{\partial net_j(n)}{\partial w_{jk}^I(n)} \\ &= \frac{\psi'_{a,b}\left[net_j(n)\right]}{a_j(n)}\left[x_k(n) + w_j^D(n)\frac{\partial \psi_{a,b}\left[net_j(n-1)\right]}{\partial w_{jk}^I(n)}\right]\end{aligned} \tag{20}$$

With initial conditions:

$$\frac{\partial \psi_{a,b}(0)}{\partial w_{jk}^I(0)} = 0 \tag{21}$$

The translation coefficient of the j-th wavlon in hidden layer can be adjusted according to:

$$\begin{aligned}b_j(n+1) &= b_j(n) + \eta\frac{\partial E(n)}{\partial b_j(n)} \\ &= b_j(n) + \eta e(n)\frac{\partial y(n)}{\partial b_j(n)} \\ &= b_j(n) - \eta e(n) w_j^O(n)\psi'_{a,b}[net_j(n)]\frac{1}{a_j(n)}\end{aligned} \tag{22}$$

The dilation coefficient of the j-th wavlon in hidden layer is updated as follow:

$$\begin{aligned}a_j(n+1) &= a_j(n) + \eta\frac{\partial E(n)}{\partial a_j(n)} \\ &= a_j(n) + \eta e(n)\frac{\partial y(n)}{\partial a_j(n)} \\ &= a_j(n) - \eta e(n) w_j^O(n)\psi'_{a,b}[net_j(n)]\frac{net_j(n) - b_j(n)}{a_j(n)^2}\end{aligned} \tag{23}$$

6. Testing the Proposed Method

The parameters of the proposed NNs have been optimized on the test data based on try and error method. **Figure 3** shows the approximation error values using RWNN for 900 data.

The **Table 1** shows four significant statistical characteristics of approximation error including maximum, minimum, average, and RMS for the approximation of GPS GDOP for 900 test data using RWNN.

The approximation with RNN and WNN has been done and the comparative results have been shown in **Table 2**. Root Mean Square (RMS) was used to evaluate approximations results [10]. RMS value is computed using:

$$\mathrm{RMS} = \sqrt{\frac{1}{T}\sum_{i=1}^{i=T}\left(\mathrm{GDOP}_{\mathrm{Real}} - \mathrm{GDOP}_{\mathrm{NN}}\right)^2} \tag{24}$$

where T is number of tests.

As it is visible over this Table, the RWNN is more

Table 1. Maximum, minimum, average and RMS error for the approximation of GPS GDOP for 900 test data determined using RWNN.

Parameters	Value
Maximum	3.5928
Minimum	-3.4024
Average	0.0544
RMS	**0.4590**

Table 2. Comparison of RNN, WNN and RWNN performance for GPS GDOP approximation.

NN Type	RNN	WNN	RWNN
RMS Value	0.5338	0.5172	0.4590

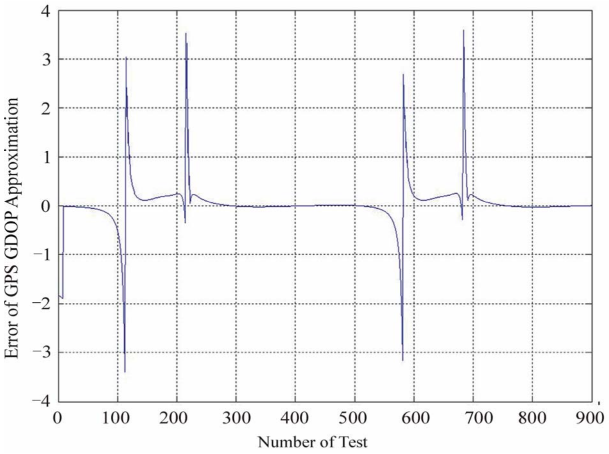

Figure 3. The approximation error values of GPS GDOP determined based on RWNN for 900 data with (4, 3, 1) structure.

Table 3. Comparison CPU time of classical method and NN approach for GPS GDOP approximation.

Model Name	CPU Time [msec.]
Matrix inversion method	1.080
NN approach	0.029

efficiency in comparing with RNN and WNN; this is because of the RMS approximation error shortage over them.

Table 3 presents the comparison CPU time of classical method and NN approach for GPS GDOP approximation. The simulation results demonstrate that NN approach is accurate and faster than classical method.

7. Conclusions

In this paper, the rapid and precise calculation of GPS GDOP using RWNN has been studied for the selection of an appropriate subset of navigator satellites. The method of NNs is a realistic computing approach used for the calculation of GPS GDOP without any need to inverse matrix, which imposes a huge computing load on the processor of the navigator. The performance of the proposed NN has been studied on the test data of the paper. The results show that the proposed method is fully capable to select an optimal subset of GPS satellites with the best geometric configuration. The results of simulation show that the efficiency of RWNN is better than RNN and WNN.

8. References

[1] R. Yarlagadda, I. Ali, N. Al-Dhahir and J. Hershey, “GPS GDOP Metric,” *IEE Proceedings—Radar, Sonar and Navigation*, Vol. 147, No. 5, 2000, pp. 259-264. doi:10.1049/ip-rsn:20000554

[2] M. Zhang and J. Zhang, “A Fast Satellite Selection Algorithm: Beyond Four Satellites,” *IEEE Journal of Selected Topics in Signal Processing*, Vol. 3, No. 5, 2009, pp. 740-747. doi:10.1109/JSTSP.2009.2028381

[3] D. Simon and H. El-Sherief, “Navigation Satellite Selection Using Neural Networks,” *Journal of Neurocomputing*, Vol. 7, 1995, pp. 247-258. doi:10.1016/0925-2312(94)00024-M

[4] M. R. Mosavi, “High Performance Methods for GPS GDOP Approximation Using Neural Network,” *Journal of Geoinformatics*, Vol. 4, No. 3, 2008, pp. 9-16.

[5] D. J. Jwo and C. C. Lai, “Neural Network-Based Geometry Classification for Navigation Satellite Selection,” *Journal of GPS Navigation*, Vol. 56, No. 2, 2003, pp. 291-304. doi:10.1017/S0373463303002200

[6] M. R. Mosavi, “GPS Receivers Timing Data Processing using Neural Networks: Optimal Estimation and Errors Modeling,” *Journal of Neural Systems*, Vol. 17, No. 5, 2007, pp. 383-393. doi:10.1142/S0129065707001226

[7] B. W. Parkinson, “Global Positioning System: Theory and Applications,” *The American Institute of Aeronautics and Astronautics*, Vol. 1, 1996.

[8] D. J. Jwo and C. C. Lai, “Neural Network-Based GPS GDOP Approximation and Classification,” *Journal of GPS Solutions*, Vol. 11, No. 1, 2007, pp. 51-60. doi:10.1007/s10291-006-0030-z

[9] M. R. Mosavi, “An Adaptive Correction Technique for DGPS using Recurrent Wavelet Neural Network,” *IEEE Conference on Systems, Man, and Cybernetics*, 2007, pp. 3029-3033. doi:10.1109/ICSMC.2007.4413579

[10] O. Øvstedal, “Absolute Positioning with Single- Frequency GPS Receivers,” *Journal of GPS Solutions*, Vol. 5, No. 4, 2002, pp. 33-44. doi:10.1007/PL00012910