Error Estimation for an Integrated GPS/INS System using Adaptive Neuro-Fuzzy technique

Salam A. Ismaeel * Ahmed M. Hassan, and * Ali F. Lutfy

Received on: 14 / 8 / 2007
Accepted on: 23/11/ 2009

Abstract

Global Positioning System (GPS) and Strap down Inertial Navigation System (SDINS) can be Integrated Together To Provide A Reliable Navigation System. In This Paper, A Technique For Error Estimation In A GPS/INS System Based On A Low-Cost Inertial Measurement Unit (IMU) Is Offered. This Technique Is Composed Of Wavelet Transform (WT) And Adaptive Fuzzy System (AFS). The Wavelet Decomposition Is Used To De-Noise The Position And Velocity Components Of The GPS And INS Outputs. An AFS Is Introduced In This Paper To Estimate The Position And Velocity Errors In The Integrated System In Order To Provide Accurate Navigation Information About The Moving Vehicle.

Several Data Sets Are Processed In This Paper, Where The Simulation Results Are Based On Matlab7 Programming Language. Six AFS Networks Are Used To Process The Position And Velocity Components. The Average Error Value Per Sample Was 0.0142, 0.0443, And 0.0108 M For Position In X, Y, And Z Axes Respectively And 0.0077, 0.0223, And 0.0269 M/S For Velocity In North, East, And Down Directions Respectively

Keyword: GPS, INS, Integrated System, Wavelet multi-resolution analysis, Adaptive Fuzzy System.

الخلاصة

منظومة تحديد الموقع العالمي(GPS) ومنظومة الملاحة النطاقية من النوع الثابت(SDINS) يمكن ان تتكاملان معا لتوفير منظومة ملاحة موثوقة. في هذا البحث, قدمت تقنية لتخمين الخطأ في منظومة ال(GPS) وال(INS) مبنية على اساس منظومة (IMU) قليلة الكلفة. هذه التقنية مركبة من تقنية تحويل المويجة وتقنيةالشبكة المكيفة المضببة. حيث ان تحويل المويجة يستخدم لتقليل الضوضاء في مركبات الموقع والسرعة لمخرجات كلأ من ال(GPS) وال(INS). منظومة الشبكة المضببة المقدمة في هذا البحث لأجل تخمين الخطأ بالموقع والسرعة في المنظومات المتكاملة لغرض توفير معلومات ملاحية حول المركبة المتحركة.

عدة مجموعات من البيانات قد استخدمت في هذا البحث, حيث ان نتائج المحاكاة بنيت بأستخدام لغة البرمجة (Matlab7). ستة شبكات مضببة هي ضرورية لمعالجة مركبات السرعة والموقع. معدل قيمة الخطأ لكل وحدة هو 0.0142 , 0.0443 , و0.0108 متر للموقع بالنسبة للمركبات x,y,z على التوالي و 0.0077, 0.0223 , و0.026 متر/ثانية للسرعة للمركبات الثلاثة (north, East, Down) على التوالي.

*Control and Systems Engineering Dept.University of Technology / / Baghdad-IRAQ

1.Introduction

Since the 1940s, navigation systems, in particular inertial navigation systems (INSs), have become important components in military and scientific applications. In fact, INSs are now standard equipment on most planes, ships, and submarines [1].

SDINS technologies are based on the principle of integrating specific forces and rates measured by accelerometers and rate gyros of an Inertial Measurement Unit (IMU) fixed on the moving body [2]. On the other hand, the GPS relies on the technique of comparing signals from orbiting satellites to calculate position (and possibly attitude) at regular time intervals. But being dependent on the satellites signals makes GPS less reliable than self contained INS due to the possibility of drop-outs or jamming [3, and 4].

The combination of GPS and INS has become increasingly common in the past few years because the characteristics of GPS and INS are complementary. This paper looks for away in high quality integration where low cost inertial sensors are used to obtain improved performance.

GPS and INS both can be used for wide range of navigation functions. Each has its strengths and weaknesses as illustrated in table (1).

2. Problem Statement

Many researches investigated and developed the INS/GPS integration systems using different approaches such as Kalman filtering, accelerometer and gyro calibration and compensation, also other methods are described in [5, and 6]. While this paper is different

in handling the deficiency in navigation systems utilizing the adaptive fuzzy system.

In general, GPS/INS integration employing AFS provides several advantages if compared to Kalman filtering. A comparison between both techniques is given in table (2).

A new method will be introduced in this paper based on AFS to fuse the outputs of INS and GPS and provide accurate positioning information and velocity for the vehicle. In addition, this paper suggests a wavelet multiresolution analysis (WMRA) algorithm to analyze and compare the INS and GPS outputs at different resolution levels before processing them by the AFS module during either the training or testing phases.

3. Adaptive Fuzzy System Structure [7, 8, 9]

Different interpretations for the fuzzy IF-THEN rules result in different mappings of the fuzzy inference engine, also there are different types of fuzzifier and defuzzifier. Several combinations of the fuzzy inference engine, fuzzifier, and defuzzifier may constitute useful fuzzy logic system. If the fuzzy logic system can be represented as a feed forward network, then the idea of back propagation training algorithm can be used to train it.

The most useful class of defuzzifier is the center average of the form:

$$f(\underline{x})=\frac{\sum_{j=1}^{M} y_j(\mu_{F_j}(y_j))}{\sum_{j=1}^{M}(\mu_{F_j}(y_j))} \quad \ldots(1)$$

Where M is the number of fuzzy IF-THEN rules, y_j is the center of fuzzy set f_j, that is, a point in the universe of discourse V at which $\mu_{F_j}(y)$ achieves its maximum value, and $\mu_{F_j}(y)$ is given by a product inference engine, since the product operator retains more information than MIN operator when implementing the fuzzy AND because the latter scheme only retains one piece of information whereas the product operator combines n-pieces. Also, using product operator generally gives a smoother output surface, a desirable attribute in modeling and control systems.

Hence, equation (1) becomes:

$$f(\underline{x})=\frac{\sum_{j=1}^{M} y_j(\prod_{i=1}^{n}\mu_{F_{ij}}(x_i))}{\sum_{j=1}^{M}(\prod_{i=1}^{n}\mu_{ij}(x_i))} \quad \ldots(2)$$

where n is the number of input linguistic variables.

In order to develop training algorithm for this fuzzy logic system, the functional form of $\mu_{F_i}(x_i)$ must be specified. The bell-shaped membership function, based on the normal distribution of the grades of the membership, would be used, since this function is differentiable and can be applied when using the back propagation learning algorithm, i.e. the membership function can be given by the following equation:

$$\mu_{F_i}(x_i)=\exp\left[-\left(\frac{x_i-m_i}{\sigma_i}\right)^2\right] \quad \ldots(3)$$

where m_i and σ_i are, respectively, the center and width of the bell-shaped function of the ith input variable.

Now from equation (2) and equation (3) the overall function of fuzzy logic system can be obtained:

$$f(\underline{x})=\frac{\sum_{j=1}^{M} y_j\left[\prod_{i=1}^{n}\exp\left[-\left(\frac{x_i-m_{ij}}{\sigma_{ij}}\right)^2\right]\right]}{\sum_{j=1}^{M}\left[\prod_{i=1}^{n}\exp\left[-\left(\frac{x_i-m_{ij}}{\sigma_{ij}}\right)^2\right]\right]}$$

…(4)

This equation represents a fuzzy logic system with center average defuzzifier, product inference rule, singleton fuzzifier, and bell-shaped membership function. Wang [7] shows that this fuzzy logic system is universal approximator (i.e. capable of uniformly approximating any nonlinear function to any degree of accuracy).

Equation (4) can be represented as a feed forward neural network as shown in figure (1). This connectionist model combines the approximate reasoning of fuzzy logic into a neural network structure.

With five-layered structure of the proposed connectionist model, the basic functions of the nodes in each layer would be defined as follows:

Associated with each node in a typical neural network is an integration function which serves to combine information or activation from the other nodes.

This function X_i^1 provides the net input of the ith node in layer 1. A second action taken by each node is to output an activation value as a function of its net input:

$$O_i^1(k) = g(X_i^1(k)) \ldots (5)$$

where g(.) denotes the activation function.

The functions of the nodes in each layer of the fuzzy-neural network can be summarized as follows:

1) Input Layer

The nodes in this layer just transmit their input values directly to layer2:

$$X_1^1 = x_1 \ ,X_2^1 = x_2 \ ,\ldots\ldots,\ X_n^1 = x_n \quad \ldots(6)$$

$$O_i^1 = X_i^1 \quad \ldots(7)$$

where i=1,2,...,n and n is the number of the input linguistic variables.

2) Antecedent Layer

The output from this layer is described by:

$$O_i^2 = \mu_{F_i}(X_i^2) \quad \ldots(8)$$

where X_i^2 is the input to node i in layer2 and F_i is the linguistic label assigned to fuzzy set (small, large, etc.).

From equation (3), equation (8) becomes:

$$O_i^2 = \exp\left[-\left(\frac{X_i^2 - m_{ij}}{\sigma_{ij}}\right)^2\right] \ \ldots (9)$$

where m_{ij} and σ_{ij} are, respectively, the center and width of the bell-shape function of the ith input of the jth rule.

3) Rule Layer

The magnitude of the output from each node in this layer is dictated by the firing strength of a rule. With the proposed scheme (i.e. equation (4)), the rule nodes perform the fuzzy product operation; Therefore:

$$z_j = O_i^3 = \prod_{i=1}^{n} X_{ij}^3 \quad \ldots(10)$$

where X_{ij}^3 denotes the ith input to node j in layer 3.

4) Consequent Layer

From this layer, the upper node sums all outputs from the rule layer with action strengths (y_j) and the lower node sums those with unity strength, as shown:

$$N = O_1^4 = \sum_{j=1}^{M} y_j X_j^4 \quad \ldots(11)$$

$$D = O_2^4 = \sum_{j=1}^{M} X_j^4 \quad \ldots(12)$$

where N and D represent, respectively, the numerator and denominator of equation (4).

5) Action Layer

Only one node exits in this layer. Here the actual output would be pumped out the net,

$$f(\underline{x}) = O^5 = \frac{N}{D} \quad \ldots(13)$$

<u>3.1 Adaptive Fuzzy System Training Algorithm</u> <u>[8, 10]</u>

Based on the idea of the error back propagation algorithm, the goal is to determine a fuzzy logic system $f(\underline{x})$, in the form of equation (4), which minimizes the error function:

$$E(k) = \frac{1}{2}\sum_{j=1}^{p}[f_j(\underline{x}(k)) - d_j(k)]^2 \ (14)$$

where P is the number of outputs and $d_j(k)$ is the jth desired output at time k. Without loss of generality, Multi-Input Single-Output (MISO) fuzzy logic system was considered in this paper. A multi-output system can be decomposed into a group of single-output systems, therefore for P=1, equation (14) is reduced to:

$$E(k) = \frac{1}{2}\left(f(\underline{x}(k)) - d(k)\right)^2 \ldots(15)$$

According to equation (4), if the number of rules is M, then the problem becomes training the parameters y_j, m_{ij}, and σ_{ij} such that $E(k)$ is minimized.

Based on the back propagation training algorithm, the iterative equations for training the parameters y_j, m_{ij}, and σ_{ij} are:

$$y_j(k+1)=y_j(k)-\eta(f(\underline{x}(k))-d(k))\frac{1}{D}O_j^3 \quad (16)$$

$$m_{ij}(k+1)=m_{ij}(k)-2\eta\frac{Z_j}{D}(f(\underline{x}(k))-d(k))\cdot (y_j(k)-f(\underline{x}(k)))\left(\frac{X_i^2(k)-m_{ij}}{(\sigma_{ij})^2}\right) \quad (17)$$

$$\sigma_{ij}(k+1)=\sigma_{ij}(k)-2\eta\frac{Z_j}{D}(f(\underline{x}(k))-d(k))\cdot (y_j(k)-f(\underline{x}(k)))\left(\frac{(X_i^2(k)-m_{ij})^2}{(\sigma_{ij})^3}\right) \quad (18)$$

where η is the learning rate. Equations (16), (17), and (18) perform an error back propagation procedure.

4. Gps/Ins System Intgration Using Adaptive Fuzzy-Wavelet Techniques

This system is able to incorporate qualitative and quantitative information. The system was represented as a feed forward neural network. Supervised linear back propagation learning algorithm was applied to adapt the fuzzy parameters. The fuzzy system with the training algorithm is called the adaptive fuzzy system (AFS).The proposed adaptive fuzzy-wavelet techniques to be applied here consists of three phases:

4.1 Construct Ins/Gps Error Signal Phase

In this phase, an INS/GPS error signal is constructed where the WMRA algorithm is used to process the GPS and INS data of 15 trajectories for each position and velocity component and to output a GPS/INS error signal associated with each trajectory, i.e. for each couple of GPS data and INS data, the WMRA algorithm constructs a GPS/INS error signal. These error signals will be compared with the output of the AFS networks, i.e. they are used as target outputs to the AFS networks. Where the WMRA will be discussed below.

4.1.1 Multi-Resolution Analysis

Scaling a wavelet simply means stretching or compressing it. The smaller the scale the more compressed the wavelet is, while the larger the scale the more stretched the wavelet is. Therefore, lower scales allow for analysis rapidly changing details (high frequency components) [11]. Similary, higher scales allow for analysing slowly changing features (low frequency components). The low frequency contents of the signal are usually the most important part of the signal that identifies the signal itself and are capable of providing a very good approximation about the signal [11]. The approximations correspond to the high scale low frequency part. On the other hand, the high frequency contents carry few details about the signal [11]. The details correspond to the low scale high frequency part.

WMRA is therefore based on the approximation and details provided using WT. WMRA decomposes the signal into various resolution levels. The data with coarse resolution contain information about the low frequency components and retain the main features of the original signal. The data with fine resolution retain information about the high frequency components [12].

In general, a space V_j can be separated into two sub-spaces: a subspace V_{j-1} (approximation) and a space W_{j-1} (detail) which is just the difference of these two spaces [12]. If this process is iterated, successive approximations will be decomposed in turn, so that one signal is broken into many fine resolution components. The original signal can then be reconstructed from the sum of the final approximation component and the detail components of all levels.

Consider j and k to be the dilation (scaling) index and the translation (shifting) index, respectively. Each value of j means analyzing different resolution levels of the signal. The mathematical procedure of WMRA for either the INS or GPS output signals is as follows:

(1) For an input signal $x(n)$, calculate the approximation coefficient $a_{j,k}$ at the jth resolution level as follows [11]:

$$a_{j,k} = 2^{(-j/2)} \sum_{n} x(n) \phi(2^{-j} n - k) \quad (19)$$

Where $\phi(n)$ is called the scaling function. Scaling functions are similar to wavelet functions except that they have only positive values. They are designed to smooth the input signal (i.e. seeking the signal approximation). They work in the signal in a way similar to averaging the input signal $x(n)$. The scaling function is applied to the input signal to determine the approximation. This operation is equivalent to low pass filtering.

(2) The approximation of $x(n)$ at the jth resolution level is then computed as [11]:

$$x_j(t) = \sum_{k=-\infty}^{\infty} a_{j,k} \phi_{j,k}(t) \ldots (20)$$

Calculate the detail coefficient $d_{j,k}$ at the jth resolution level [11]:

$$d_{j,k} = \sum_{n} x(n) \psi_{j,k}(n) \ldots (21)$$

it can be noticed that the wavelet function is used in calculating the detail coefficient. Wavelet functions $\psi_{j,k}(n)$ are designed to seek the details of the signals. They work in the signal in a way similar to differentiation (giving the difference between two consecutive samples of the input x (n)). This is why a wavelet function $\psi_{j,k}(n)$ consists of positive and negative parts. The detail function will be applied to the input signal to determine the details. This operation is equivalent to high-pass filtering.

(3) The detail of x (n) at the jth resolution level is then computed as follows [11]:

$$g_j(n) = \sum_{k=-\infty}^{\infty} d_{j,k} \psi_{j,k}(n) \ldots (22)$$

The above four steps are repeated for the j+1 resolution level but by using the approximation $x_j(n)$ obtained in step 2.

The original signal x (n) can be reconstructed using all the details obtained during the decomposition process at all resolution levels [11]:

$$x(n)=\sum_{j=-\infty}^{\infty}g_j(n)=\sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty}d_{j,k}\psi_{j,k}(n) \quad (23)$$

The above equation implies that one has to process the original signal at an infinite number of resolutions, which is impractical. Alternatively, the analysis can stop at the jth resolution level and the signal can be reconstructed using the approximation at that level and all the details, starting from the first resolution level until the Jth level. The following equation presents this procedure [11]:

$$x(n)=\sum_{k=-\infty}^{\infty}a_{j,k}\phi_{j,k}(n)+\sum_{j=1}^{J}\sum_{k=-\infty}^{\infty}d_{j,k}\psi_{j,k}(n) \quad (24)$$

The first term represents the approximation at level J and the second term represents the details at resolution level J and lower. In conclusion, WMRA builds a pyramidal structure that requires an iterative application of scaling and wavelet functions as low-pass (LP) and high-pass (HP) filters, respectively. These filters initially act on the entire signal band at the high frequency (lower scale values) first and gradually reduce the signal band at each stage.

The WMRA is applied to both the INS and GPS position and velocity output components (X, Y, and Z) with individual analysis provided for each component. The INS monitors the linear acceleration and angular velocities of the vehicle with minimum time delay. For short time intervals, the integration of acceleration and angular rate measurements results in an extremely accurate velocity, position and altitude with almost no noise or time lags. However, because the INS outputs are obtained by integration, and the fact that the measurements contain residual bias errors from both the gyroscopes and the accelerometers, they drift at low frequencies. To obtain very accurate outputs at all frequencies, the INS is updated periodically using GPS positions and/or velocities, which complement the INS output in an ideal way. Therefore, as shown in figure (2), the WMRA technique determines the differences between the INS and GPS position and velocity outputs after comparing the corresponding position or velocity components at several resolution levels. These differences represent, in general, the INS errors, which are used to correct for the INS outputs during GPS outages. This means that the proposed navigation system will rely on the GPS position or velocity components until the GPS signal is blocked. Whenever the GPS signal is available, the GPS position or velocity component is compared to the corresponding INS position or velocity component and the corresponding position or velocity error is compared. Optimal estimation and modeling of this error signal is performed by AFS, which is discussed previously. It should be highlighted that separate WMRA of the form shown in figure (2) is designed for each position and velocity component.

In this paper, the comparison between the INS and GPS position and velocity outputs at three resolution levels was adequate. In the wavelet domain, using the DWT, the wavelet coefficients that represent one of the INS position or velocity components in the three decomposition levels are:

$$C_{INS} = [|cA_{i1}|cD_{i1}|cA_{i2}|cD_{i2}|cA_{i3}|cD_{i3}|] \quad (25)$$

The corresponding wavelet coefficients of the GPS position or velocity component are represented as:

$$C_{GPS} = [|cA_{g1}|cD_{g1}|cA_{g2}|cD_{g2}|cA_{g3}|cD_{g3}|] \quad (26)$$

In fact, *cA* and *cD* shown in the above two equations are the approximation and the detail coefficients determined in steps 1and 3 of the WMRA procedure illustrated previously.

By subtracting the wavelet coefficients of each of the GPS position and velocity from the corresponding wavelet coefficients of each of the INS position and velocity outputs, the wavelet coefficients of the error signals can be extracted as:

$$E = [|cA_{e1}|cD_{e1}|cA_{e2}|cD_{e2}|cA_{e3}|cD_{e3}|] \quad (27)$$

The error signal can then be reconstructed from the wavelet coefficients obtained in above equation. The error signal can be smoothed by neglecting the highest frequency band (the band reconstructed from the detail coefficients cD_{e1}) from the reconstructed signal. This band contains the distortions in the position and velocity components.

De-noising of the INS and/or GPS outputs can be performed within the WMRA process. This is implemented by thresolding the details of each position and velocity component, which contain the high frequency components. The thresolding procedure allows for cutting off some of the noise in the error signal and improving its signal-to-noise ratio so that it can be efficiently modeled using AFS. In this paper, soft thresholding is applied only to the detail coefficient of the first decomposition level. The thesholding procedure is standard and can be reviewed in [11, 12].

<u>4.1.2 AFS Networks Training Phase</u>

The next step is the training of the AFS networks (which is done while the satellite signal is available). Six networks are used to handle each one of the position and velocity components separately. The inputs to each network are the INS data (position or velocity component) and the instantaneous time (the time is counted once the system is turned on); the output of each network is the estimated INS error for the input component (as shown in figure (3.a)).

The error resulted from comparing the network output and the GPS/INS error signal is fed to the network which adjusts its parameters in a way to minimize the mean square value of the error. The parameters of the AFS network that are computed during the training phase are m, y and σ. These parameters are updated according to equations (16), (17), and (18). The computations of these parameters are repeated until the optimal values are achieved which correspond to the minimum mean square error. The optimal values of m, y, and σ reached at the end of the training phase are saved to be used later in the testing phase.

As mentioned, each component of position and velocity has its own network. To start the training, the networks need to be initialized with the number of epochs, the value of the learning rate, the number of fuzzy rules (M), and the parameters (m, y, and σ). These initial values which are shown in table (3) are selected by trial-and-error. Appropriate selection of the initial values ensures good performance of the networks and converging to a minimum error value.

After the training process being completed for all components, the networks are now ready to work in the testing mode.

4.1.3 Testing Phase

The final step in the AFS-wavelet method is the testing phase. After the training is completed, the network is ready to work in the testing mode. The parameters of the networks are modified during the availability of the satellite signal, i.e. in the training phase. In the case of satellite signal being blocked, the networks will use the latest modified parameters saved from the training phase to perform the prediction process.

Figure (3.b) shows the operation of the networks in the testing mode. It provides a prediction of the INS error based on the INS data and the particular time instant provided at the input.

In this paper, the training of the networks was started by an attempt to use 13 GPS/INS error signals, in each of the position and velocity networks, from the 15 trajectories mentioned previously (the other two were used in the testing phase).These 13 trajectories were chosen randomly (they had different shapes) and were used together in the training process, i.e. one trajectory after the other. Each network was implemented many times and each time the initial values were changed, in an attempt to get better performance. This process was repeated for all position and velocity components and for all trajectories.

Figure (4) shows the MSE for all networks after 1000 epoch. The initial values used to obtain these results are listed in table (3). As stated early, these values are obtained by trial-and-error.

Figure (5) shows the error between the GPS/INS error (desired output) and the estimated INS error (actual output) for all networks.

Figure (6) shows the error between the true INS data obtained from AFS networks and from INS algorithm.

5. Conclusions

The following points summarize the main conclusions of this paper:

1. In this paper, a reliable navigation system is made by combining the qualities of GPS, INS, and adaptive hybrid fuzzy system.
2. The wavelet analysis was beneficial in filtering out the noise and disturbances that may exist at the INS and GPS outputs. In addition, it provides the advantage of comparing the INS and GPS position and velocity components at different levels of resolution.
3. The process of selecting the initial values of the parameters (m, y, and σ), number of rules, and value of the learning rate is done through a trial-and error procedure and determining the

appropriate setting for one trajectory may need several attempts; therefore, handling several trajectories separately can be a very long process whereas when these trajectories are handled together, one after the other the process of selecting the appropriate initial values is done only one time.

4. The long procedure of trial-and-error in finding the optimal number of layers in the network, the number of nodes in each layer, and the activation function in each node, which exists in ANN, has been avoided in this paper by using the AFS with its constant structure.
5. The advantage of using a group of trajectories in the training process is that the AFS network can continue in giving estimation of the INS error if changes happen in the specified trajectory of a vehicle.

6. References

1. Farrell J. and Barth M., *The Global Positioning System and Inertial Navigation*, McGraw-Hill Companies, Inc. 1999.
2. David H. and John L., *Strapdown Inertial Navigation Technology*, Michael Faraday House, 2004.
3. Wellenhof, B. H., H. Lichtenegger, and J. Collins, *GPS Theory and Practice*, by Springer-Verlag / Wien, 2001.
4. Mohinder S., Lawrence R. and Angus P., *Global Positioning Systems, Inertial Navigation, and Integration*, John Wiley & Sons, Inc. 2001.
5. Chiang K.,”INS/GPS Integration Using Neural Networks for Land Vehicular Navigation Applications”, Ph.D Thesis. University of Calgary, Geomantic Engineering Dept., 2005.
6. Ismaeel S. A., "Design of Kalman Filter for Augmenting GPS to INS Systems", Ph.D. Thesis. Computer Engineering Dept., College of Engineering, Al-Nahrain University, 2003.
7. L. X. Wang, "Adaptive Fuzzy Systems and Control: Design and Stability Analysis", Prentice-Hall, Inc., 1994.
8. Ismaeel S. A. and Al-Jebory K. M., "Adaptive Fuzzy System Modeling", Eng. Technology, No. 4, Vol. 20, 2001.
9. Earl C.,*The Fuzzy Systems*, by Academic Press. 1999.
10. El-Gizawy M., El-Sheimy N., Taha M. M. R. and Noureldin A., "Neuro-Fuzzy System for GPS/INS Integration". The European Navigation Conference, 2004.
11. Burrus R., Gopenath A., and Guo H., *Introduction to Wavelet and Wavelet Transform*, A primer. Upper Saddle, NJ (USA), Prentice Hall, Inc. 1998.
12. Goswami J. C. and Chan A. K., Fundamentals of wavelets: theory, algorithms, and applications, John Wiley &sons, Inc., 605 Third Avenue, New York, NY 10158-0012, (212) 850-6011, 1999.

Table (2): Comparison between AFS and Kalman filtering GPS/INS Integration System.

	Kalman filtering	AFS
Model dependence	Mathematical model; deterministic model + stochastic model	Empirical and adaptive model
A priori Knowledge	Required (mainly Q and R matrix)	Not required
Sensor dependence	Re-design of Kalman filter parameters is needed for different systems	An adaptable, platform and system independent algorithm
Linearity	Linear processing	Nonlinear processing

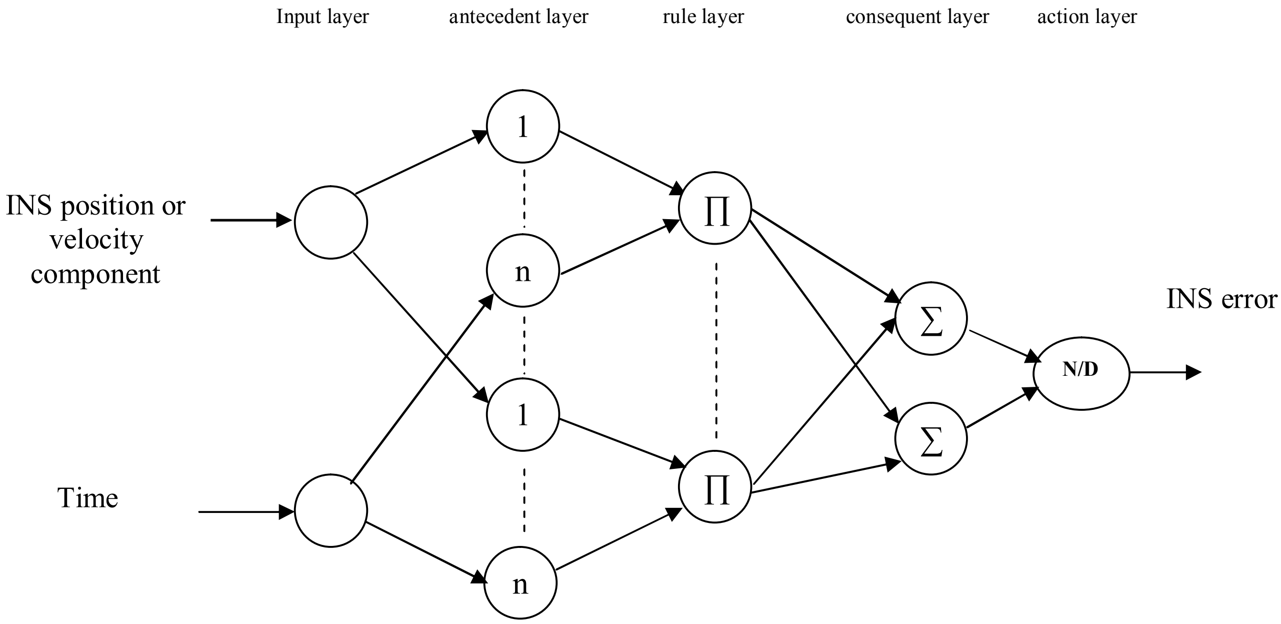


Figure (1): The Architecture of an Adaptive Fuzzy System network for each component of position (X, Y, and Z axis) and velocity (North, East, and Down).

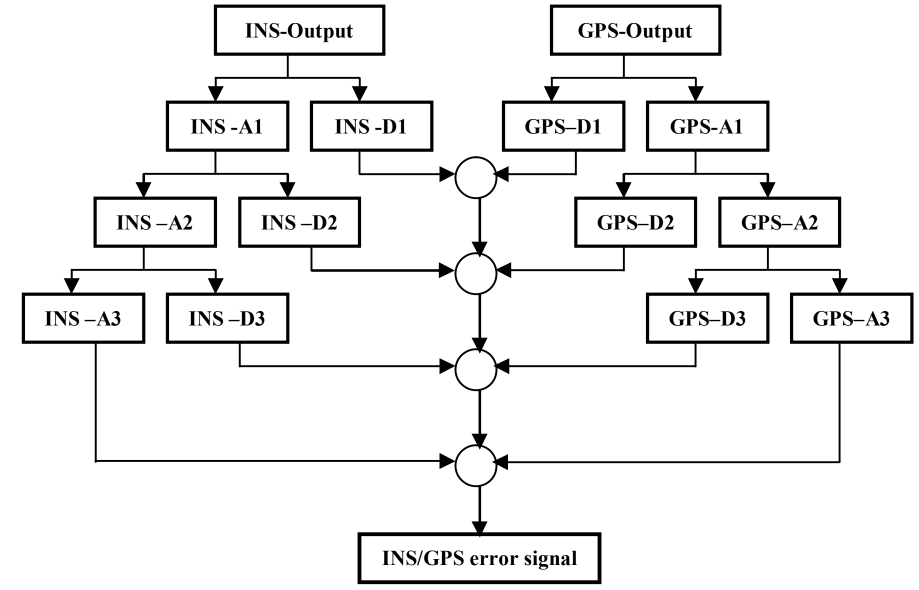
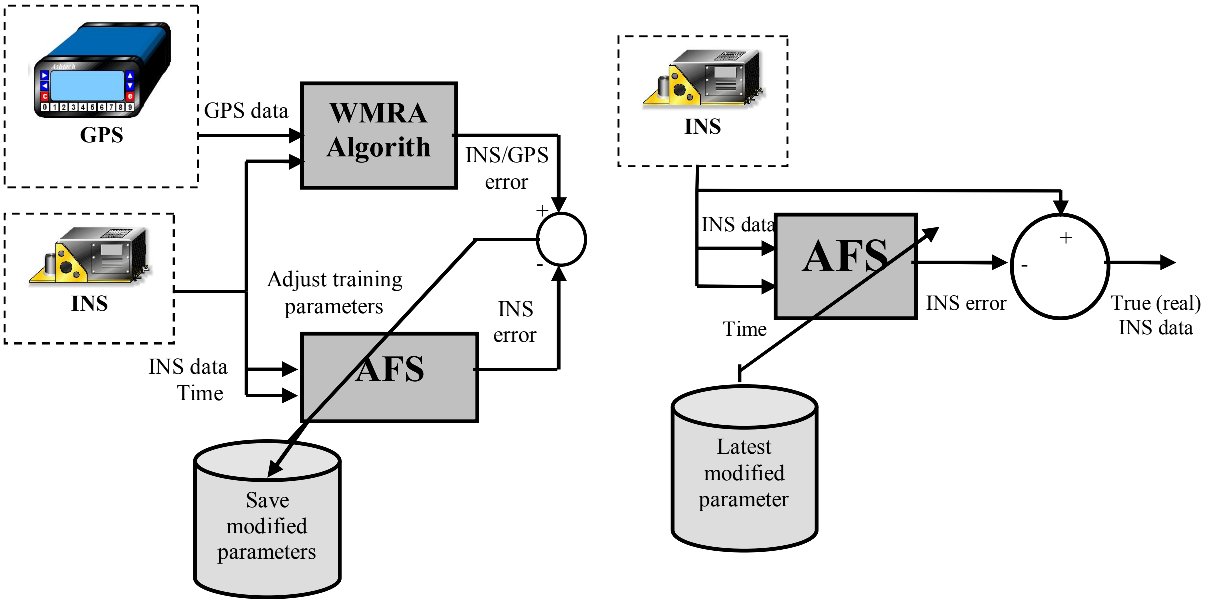


Figure (2): WMRA of the INS and GPS signals for computation of the error signal.

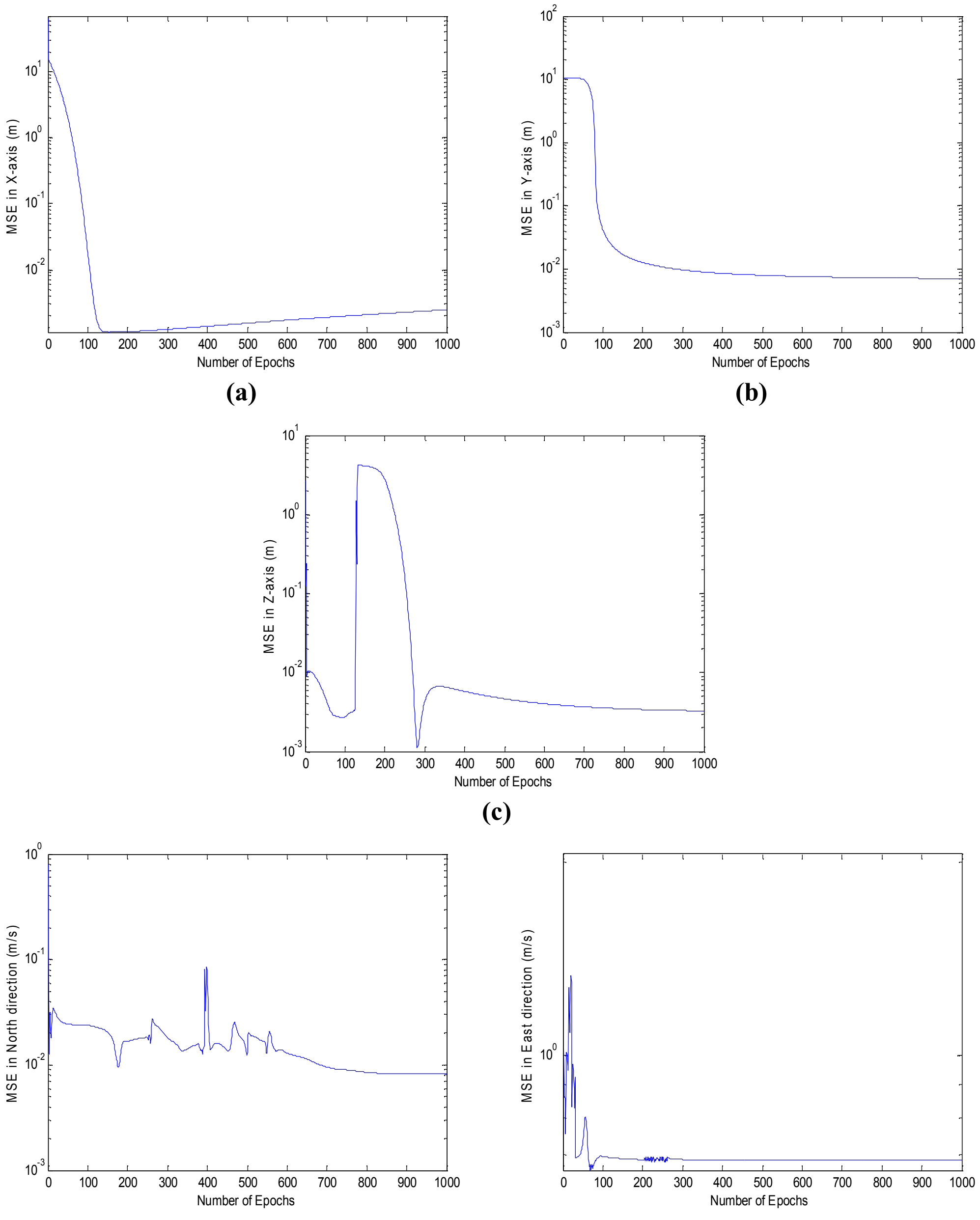


(a) **(b)**

Figure (3): Block diagrams of the AFS in (a) Training phase and (b) Testing phase.

Table (3): Initial values of the learning parameters for the six networks.

		Position			Velocity		
		X-axis	Y-axis	Z-axis	North	East	Down
Initial Values	m	[0, 15]	[-1, 1]	[-1, 1]	[0, 1]	[-2, 2]	[-1, 1]
	y	[0, 15]	[-5, 5]	[-2, 2]	[0, 1]	[-1, 1]	[-5, 5]
	σ	[0, 15]	[0.001, 1]	[0.01, 0.3]	[0, 1]	[-1, 1]	[0.001, 1]
	M	12	12	12	12	12	12
	Learning rate	0.4	0.4	0.4	0.4	0.4	0.4



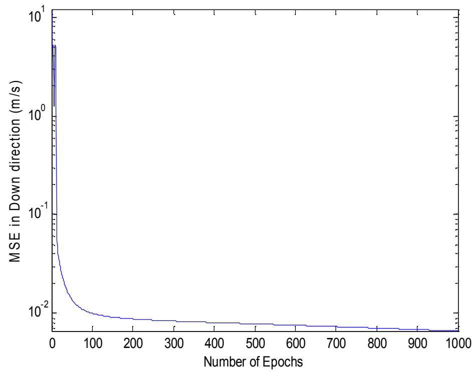
(a)

(b)

(c)

(d)

(e)



(f)

Figure (4): Mean Square Error for position in (a) X-axis, (b) Y-axis, (c) Z-axis and velocity in (d) North, (e) East, (f) Down directions.

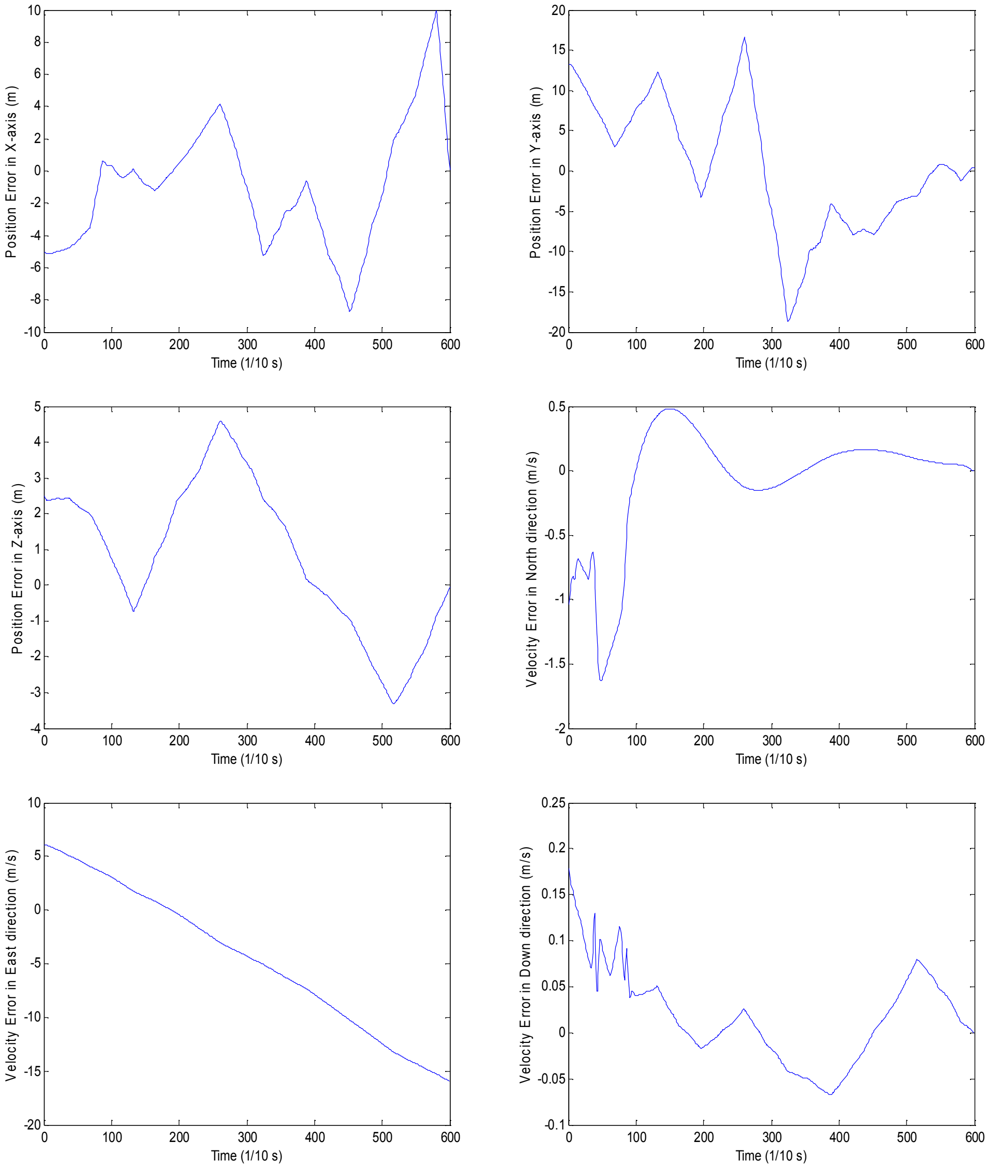


Figure (5): Error between desired and actual outputs of the AFS networks for position and velocity components.

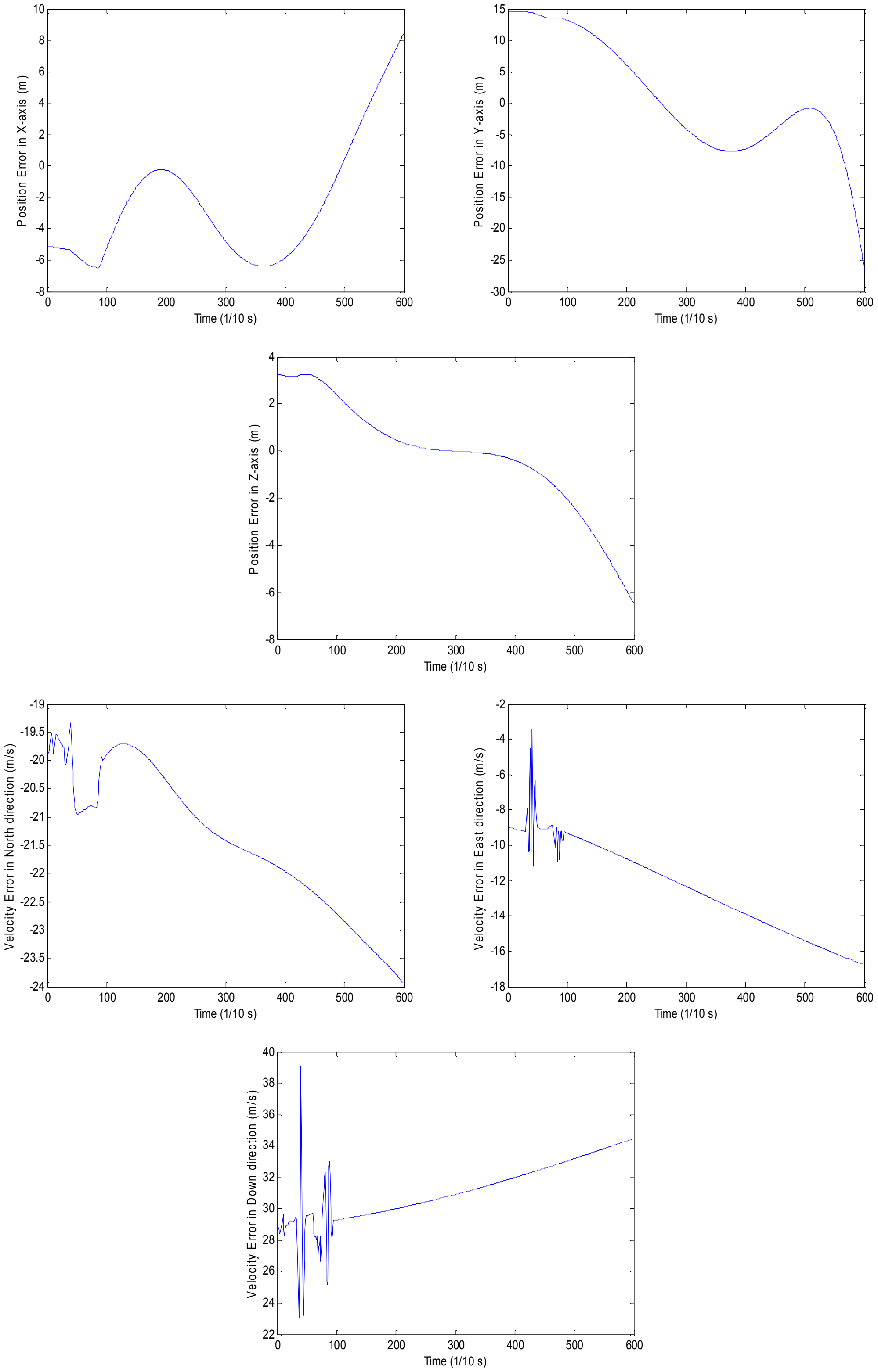


Figure (6): Error between True (Real) INS data from INS algorithm and AFS networks for all components of position and velocity.