DEVELOPMENT OF THE AUTOMATED SYSTEM ON THE BASIS OF NEURAL NETWORKS FOR FLIGHT SAFETY OF AIRCRAFT

Lukyanov A.V., Bragin A.A.
Siberian Aeronautical Research Institute named after S.A. Chapllygin, Novosibirsk
barguzin@ngs.ru; shurik_br@ngs.ru

Keywords: *Neural Network, Elman, Hopfield, ACS.*

Abstract

The concept is proposed and a prototype model of the automated system to ensure the aircraft safety based on the interaction of associative and recurrent neural networks is developed. Architectures of the used Elman and Hopfield neural networks are modified. Preliminary results confirming the correctness of the chosen concept are given.

1 Introduction

The high level of aviation accidents in the state of the Russian Federation is one of the main factors affecting the readiness of aircraft to meet its objectives and components of a threat to national security state. In the medium term aviation accidents remain one of the major challenges of sustained development and operation of state aircraft. Further increase of accidents will inevitably lead to a decrease in the motivation of citizens of the Russian Federation to the implementation of aviation, including flight operations, reduce combat readiness and combat capability of the state of aviation, as well as to reduce the export potential of Russia's military aviation technology, dual-purpose and special.

The prediction of dangerous situations is one of the major goals of artificial intelligence aircraft. The second important goal is to issue recommendations on the optimal care of the pilots from the border zone or automatic withdrawal from them. But in order to resolve two targets described above, it is necessary to carry out quality and timely forecasting of flight situation. Evaluating the effectiveness of the overall strategy for the use of neural network algorithms in onboard automated real-time and get the best practical results achieved in the process of fine-tuning of the system based on full-scale experiment. The realization of neural networks can be achieved by sequentially increasing take full advantage of the different types of neural network algorithms: the creation of multi-level hierarchical structure, building systems and structural elements, etc. Meanwhile, despite the abundance of research on the use of neural networks (NN) [1],[2],[3], the concept of criteria and algorithms for estimating the operation of onboard systems, proximity to the aircraft in a dangerous situation requires the formulation of neural network improvements associated with the use of more efficient signal processing algorithms, to model an aircraft.

2 Concept and Realization of the Automated System

Safety issues accidental withdrawal of aircraft on the boundaries of operating conditions due to an error of piloting, the impacts of external disturbances and failures of development of avionics solved intelligent systems preventing dangerous situations aircraft and control systems and alarm failures avionics. The main objective of the developed automated system - ensuring safety of flight of the aircraft through the display prompts warning to the pilot (in the form of graphics, audio, encryption, and textual information printed to the screen or a speaker at the approach of a dangerous situation) and an automatic withdrawal from a dangerous situation if the pilot does not respond to warnings.

To achieve the above goal, developed the concept of an automated control system, which includes three main elements: a system of collecting information, the system of hazard prediction and control system aircraft.

2.1 The System of Collecting Information

Collection system is designed to measure the parameters of the aircraft, converted into electrical analog, digital, analog-digital and digital signals and record the results (in binary, text and other data types) to a flash drive, external hard drive or solid-state hoarder.

The system allows you to collect the following types of signal transduction transducers flight parameters, signals satellite navigation system (SNS) and inertial navigation system (INS) and the matching devices:

- analog-digital and digital to analog conversion voltage and direct current(DC) ranges;
- signal conversion single commands;
- the transformation frequency signals up to 12.5 MHz;
- signal conversion SNS and INS.

In the configuration of the system has already composed sizes up to 310 x 190 x 170 mm and weight up to 3 kg (depending on the number of connected cards). Power can be supplied from a DC network voltage from 10 to 36 V or AC 220 V. Built-in analog-digital converters allow the connection of different types of transducers flight parameters, as a single-ended and two-wire analog signals. Input voltage and current analog signals can be changed by hardware or software. Number of connected transducers flight parameters can be varied from 1 to 32 for one card and fold increase when installing additional expansion cards, whose mass is 200 grams. The frequency of the analog-digital conversion can be as high as 100 kHz, depending on the number of connected transducers. Boards of input signals have a built-in digital-analog converter that allows you to create complex control signals. In performance corresponds to the complex technical requirements of aircraft use: operating temperature range - from -40 to +85°C, operating load - up to 40 g.

The hardware part of the information collection system is implemented in version PC/104 (block diagram shown in Figure 1) system is a single board computer Hercules-EBX-based VIA Eden processor with an integrated module of the communication device with the object (USO) which includes:

analog input: 32 channels, 16 bit, 250 kHz maximum, single-ended or differential signal connection methods, the input voltage ranges: 0 ... 10 V, 0 ... 5, 0 ... 2.5 and 0 ... 1,25; ± 10; ± 5 ; ± 2,5; ± 1,25 V;

– analog output: 4 channels, 12 bits;

a discrete input-output: 40 channels, 5V;

counter / timers: 2 channels.

Depending on the tasks, the system is equipped with ADC standard PC/104 DMM-32X-AT. Through the development of connectors, adapters (for analog and digital signals) to the Hercules-EBX also an opportunity to connect analog switch boards AIMUX-32C(32), which significantly expanded functionality. Management boards AIMUX-32C by digital signals from the port DIO Hercules-EBX. Synchronization of time samples in the learning mode by using the assembler commands based on the timing of high-precision counter of CPU cycles.

To solve the problems of navigation and organization of external trajectory measurements, developed a system for collecting information integrated modules INS FSAS-EI-SN produced by German company iMAR. This INS is working on technology with feedback. The zero drift of the gyroscopes is less than 0.75 degrees per hour, and zero drift of accelerometers - less than 1 miligala. The frequency of issuance of the inertial data is 200 Hz.

In the operating status of the receiver unit is a Canadian firm NovAtel technology SPAN, so that all inertial and satellite measurements are synchronized in time. SPAN is a technology-based algorithm is rigidly connected combination of GNSS and INS measurements, which provides highly accurate solutions even in the temporary absence of satellite signals. Rigidly connected integration means that satellite measurements are used, even when the calculation of the coordinates only on GNSS impossible**.** In addition, the technology SPAN provides a much faster re-capture satellite signals, as a result provides more data for inertial solutions. All processing of satellite and inertial data is held receiver processor series OEMV. INS block is placed in a separate enclosure from the GNSS receiver, so the range is modular.

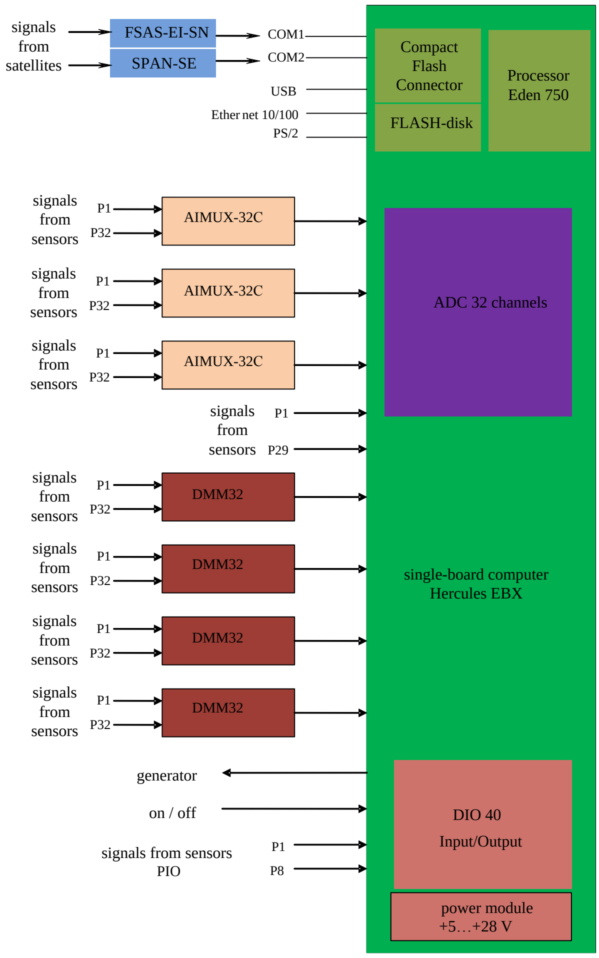


Fig. 1 Block diagram of the FID in the format of PC/104 (Hercules)

GNSS receiver SPAN-SE shows GNSS + INS raw data and solutions to several ports, or burn them onto a removable SD card. Multiple outputs for synchronizing to GPS-time input events for the labels allow easy integration into complex systems. In addition, to determine the reference direction (of course), the receiver SPAN-SE supports simultaneous operation of two antennas**.**

In order to automate the entire system is designed special software, which allows carrying out all works from the moment of internal testing and calibration dependences, to obtain the final result of treatment.

In its structure, the software consists of three subprograms:

1. The training program and configure the board MSBI;
2. Program for collecting and primary processing of information;
3. The program of secondary processing.

Programs written in C++ under the Realtime Operating System QNX Neutrino has an intuitive window interface and is based on a dialog interaction with the user.

The system of data collection is fully prepared and tested on different aircraft. The tests on the quality, reliability and noise immunity.

2.2 The Forecasting System Hazards

The purpose of this system - the prediction of the onset of a possible dangerous situation and alert the crew. For the solution of the problem, we went in the following way: splitting the flight of aircraft in the individual modes (takeoff, flight, landing) on the analysis and design criteria for each mode of the system, implementation of predictor-based Elman neural networks.

To improve the reliability and performance of neural network algorithms, and the whole idea in general, decided to split the job of the regimes of motion of the aircraft. Each of the flight data is analyzed as a separate option, issuing three possible values (flag state).

By flag-states determined by the flight mode under the decision tree presented in Figure 2. A number of values, which compares the current values of flight parameters, given the external interface system and transmitted to the module when it is loaded. The airfield takeoff is defined in the singular, whereas the landing airfields are stored in the base airfield. When the condition on the height of landing module compares the current position with a set of coordinates of the base airfield landing and in coincidence with one of them sets the selected airport, as planting a tree and moves to the next parameter of comparison.

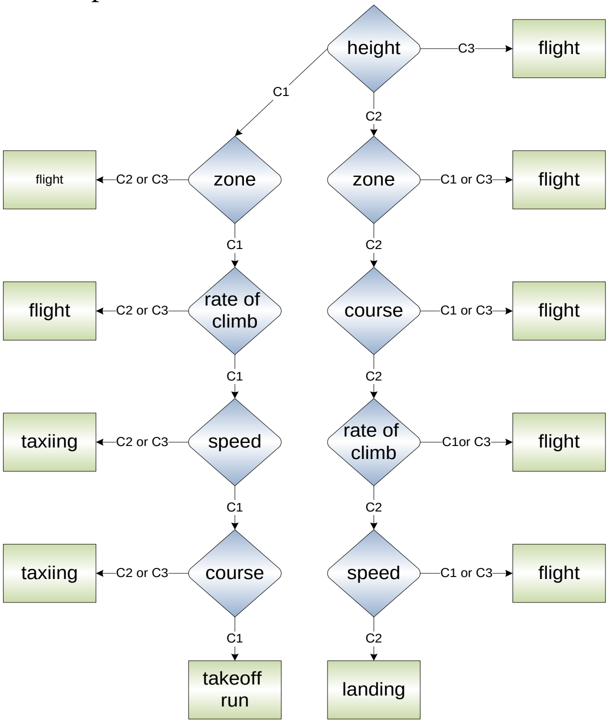


Fig. 2 Decision tree to determine the mode of flight

Thus, the module can detect four basic modes of flight:

- Taxiing;
- Take-off run;
- The flight;
- Landing;

In this decision tree is constructed in such a way that the regime was determined with a minimum number of inspections.

Having determined the flight mode, turn to the definition of dangerous situations themselves, but first will need to make the following calculations, the parameters relating to the aircraft.

First, perform calculations on the parameters for a particular aircraft: the definition of take-off weight $m_в$ (empty plane + fuel + people + cargo), the definition of the maximum front-plane alignment (tabular values for each aicraft) Determination of available long $L_{пв}$ aborted takeoff, to calculate the allowable takeoff weight for soil strength> 10 kg/cm2; determination of the rate decision V_1 velocity of the front lifting waste water $V_{пст}$, takeoff safety speed V_2 and the separation speed $V_{оТ}$; forecast parameters and verification of traffic safety conditions on the rise.

As an example, consider the rise. The whole regime is divided into stages (Fig 3).

Each stage is characterized by a set of conditions that keep track of several parameters such as speed, altitude, and others, which concludes on the correct behavior of the aircraft.

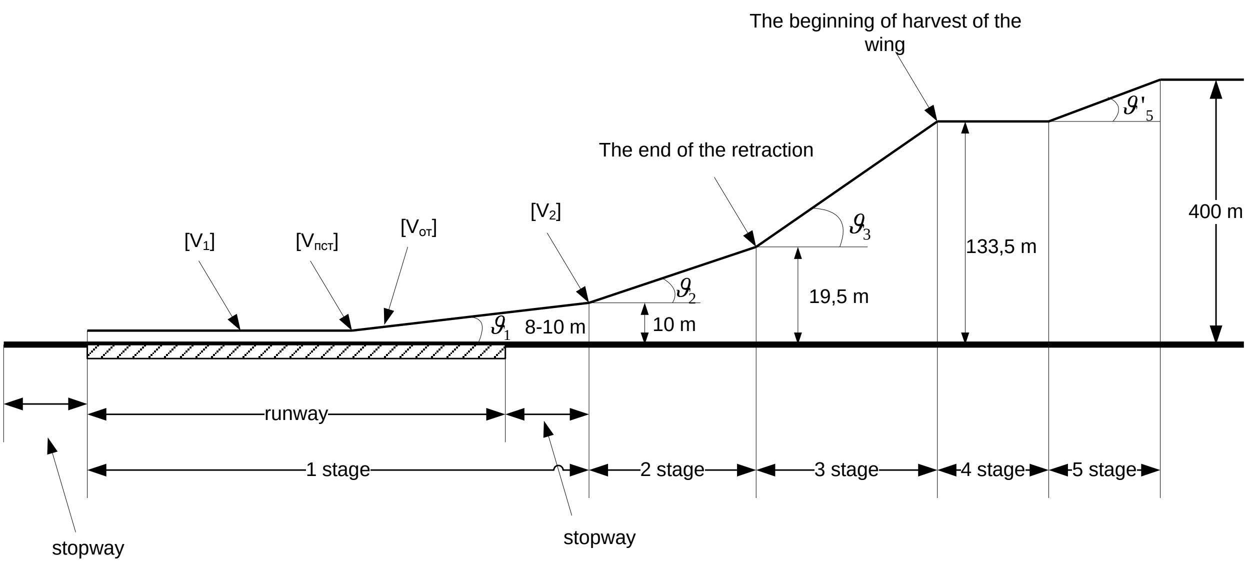


Fig. 3 Phase partitioning off of the substeps.

To improve safety, the conclusion made about the parameters taking into account the forecasted real. As neural networks are used Elman network. It was determined that more accurately the relationship between the signals of information channels are approximated by the NN with two recurrent layers. To ensure appropriate feedback Elman network commonly used sigmoid activation function. Network diagram is shown in Fig 2.

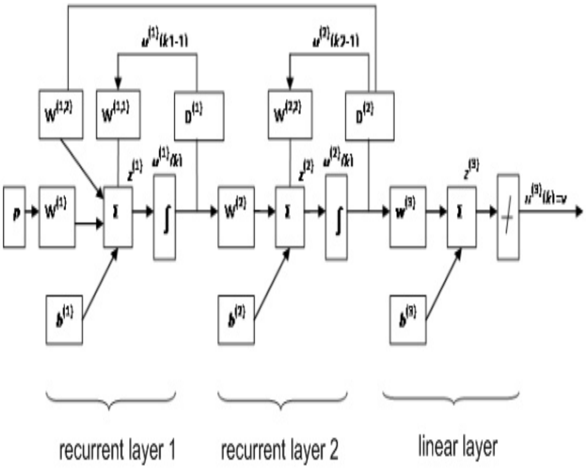


Fig. 4 Elman Network

Here p - vector of input neurons; b (1), b (2), b (3) - the displacement vector, performs the correction of sensitivity of neurons; W (1), W (2), w (3) - matrix and vector of the weights of the first and the second and third layers; W (1,1), W (1,2), W (2,2) - matrix of weights feedback first and second layers; z (1), z (2) - vector inputs for the function activation (hyperbolic tangent) of the first and second layers; z (3) - scalar input to the linear activation function, D (1), D (2) - units of delay. u (1), u (2) - Vector outputs of the first and second layers, u (3) = y-output of the network.

We believe that the output from the network and a set of values in the training set, and the entrance is a vector of size m. Description of the recurrence of layers can be written as

$$\begin{cases} z^{(1)}(k) = W^{(1,1)}u^{(1)}(k1-1) + W^{(1,2)}u^{(2)}(k2-1) + \\ + W^{(1)}u + b^{(1)},\ u^{(1)}(0) = 0, \\ u^{(1)}(k) = f\left[z^{(1)}(k)\right], \quad k = \overline{1, M}\ , \end{cases} \quad (1)$$

$$\begin{cases} z^{(2)}(k) = W^{(2,2)}u^{(2)}(k2-1) + W^{(2)}u^{(1)}(k) + b^{(2)}, \\ u^{(2)}(0) = u_0^{(2)} \\ u^{(2)}(k) = f\left[z^{(2)}(k)\right], \quad k = \overline{1, M}\ , \end{cases} \quad (2)$$

where M - total number of vectors p, consecutively received at the input; f [*]- activation function, k1, k2 - time shifts.

Line layer is described by the relations

$$\begin{cases} z^{(3)}(k) = w^{(3)}u^{(2)} + b^{(3)}, \\ u^{(3)}(k) = z^{(3)}(k). \end{cases} \quad (3)$$

We write the expression for the conversion of a single neuron in the recurrent layer

$u_i^{(1)}(k)$= f[$z_i^{(1)}(k)$], $u_j^{(2)}(k)$= f[$z_j^{(2)}(k)$], (4)

$$z_i^{(1)}(k)= \sum_{n=1}^{m} w_{in,}^{(1)}\, p_n(k) + \sum_{n=1}^{M1} w_{i,n}^{(1,1)}\, u_n^{(1)}(k1-1) + + \sum_{n=1}^{M2} w_{n,j}^{(1,2)}\, u_n^{(2)}(k2-1), \quad (5)$$

$$z_j^{(2)}(k)= \sum_{n=1}^{M1} w_{j,n}^{(2)}\, u_n^{(1)}(k) + \sum_{n=1}^{M2} w_{n,j}^{(2,2)}\, u_n^{(2)}(k2-1)$$

where $u_i^{(1)}(k)$, $u_j^{(2)}(k)$, $u_n^{(1)}(k1\text{-}1)$, $u_n^{(2)}(k2\text{-}1)$ – outputs the i-th and j-th neuron of the first and second layers at time k and k1-1, k2, respectively; $z_i^{(1)}(k)$, $z_j^{(2)}(k)$ - inputs to the activation function, M1, M2 - number of neurons in the first and second layers; $p_n(k)$ - n-th element of the entrance of the NN at time k; $w_{in,}^{(1)}$- the weight of the i-th neuron of the first layer of the n-th element of the entrance of the NN; $w_{jn}^{(2)}$ - the weight of the j-th neuron of the second layer from the n-th neuron of the first layer; $w_{i,n}^{(1,1)}$- the weight of the feedback to the i-th neuron of the first layer of the n-th neuron of the first layer; $w_{n,j}^{(1,2)}$ - the weight of the feedback to the i-th neuron of the first layer of the n-th neuron of the second layer; $w_{n,j}^{(2,2)}$ weight of the feedback to the j-th neuron of the second layer from the n-th neuron of the second layer.

If you are learning the value of inputs and outputs (outputs). Used for recurrent networks backpropagation algorithm reduces time to deploy a network of operations in a multilayer feedforward networks. Denote the set value the network output for the k-th iteration of the calculation, which coincides with the k-th readout $u_з^{(3)}(k)$. Find the square of the difference between the specified and calculated values, consistently revealing the layers of the signals from the latter to the former. In this case, the type of activation functions take the same for all neurons.

After equating the derivatives, we obtain expressions for the specification of weights.

The derivatives of the activation function are represented in the form of hyperbolic tangent:

$$\frac{d\,f^{(1)}}{d\,z_n^{(1)}}=1-[u_n^{(1)}(k)]^2; \qquad \frac{d\,f^{(1)}}{d\,z_n^{(2)}}=1-[u_n^{(2)}(k)]^2 \qquad (6)$$

In connection with the centering bias of all signals b1, b2, b3 be zero.

The initial values of weights are assumed equal d/5-0, 1, where d - a random number with uniform distribution [0, 1]. The initial values of all derivatives of the signals $u^{(1)}$, $u^{(2)}$ of all the weights take the zero.

After calculating the weights of all the available values for learning, recursive process is repeated for as long as the sum of squared errors $\sum_{k=1}^{K} \Delta u^{(3)}(k)$, where K - number of output set values will not vary from iteration to iteration for a given value of the error.

Regarding our problem, the neural network is trained for the inputs - control actions, outputs - measuring pitch and roll angles and measuring the height of about ~ 2 seconds before.

Thus, when teaching the ins and outs are reduced by the mode of the algorithm, the input (each) lags behind the output within 0.1 ÷ 1.5 s. The exact value is determined by minimizing the error of prediction.

Inputs for the NN roll: the position of the ailerons, rudder, turns left and middle engine height.

Inputs for the NN of pitch: the provisions of the elevator, ailerons, motor speed, height (not yet take into account the impact on the speed difference from the standard atmospheric conditions).

Inputs for the NN height: the position of the elevator, ailerons, motor speed.

Working the network:

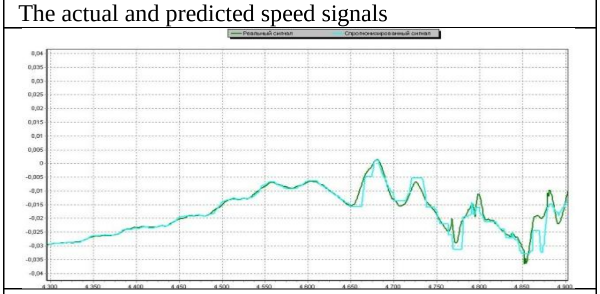
The calculated values of the network changes are compared with actual changes derived from full-time smoothed signals. If changes to the National Assembly to have the actual $3 \cdot \sigma_{uз}$, we can assume that the situation dangerous.

Mean square error changes $\sigma_{uз}$ for roll, pitch and height are determined during training as a mean square error difference between the given and calculated values of the neural network for the learning stage. In fact, it means clarification $\sigma_{uз}$ from cycle to cycle training. Cycle - a flight. Education after the flight. The search of optimal intervals backlogs input should be held earlier. Pre-processing of signals (smoothing) is performed equally in the training and operation.

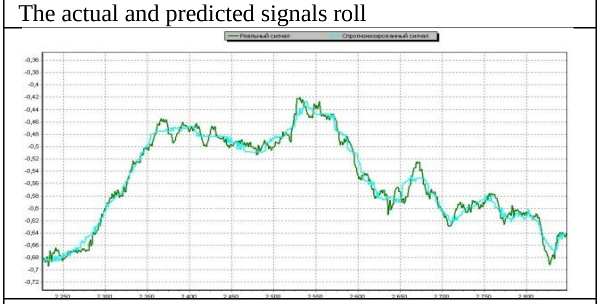
Using data obtained during the accumulation of a database, tested prediction of the algorithm with the following results(Table 1):

Table 1 - Results of the algorithm pronozirovaniya

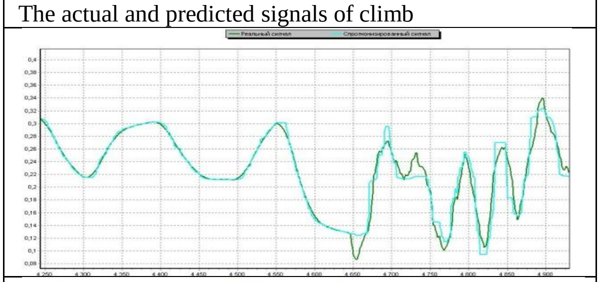
The actual and predicted speed signals



The actual and predicted signals roll



The actual and predicted signals of climb



Prediction signal   Real signal

2.3 The Control System Aircraft

As the neural structures for controlling the aircraft, as noted above, the choice was made in favor of an associative Hopfield neural network. This choice is explained by the fact that the Hopfield network is a self-learning network with feedback, respectively, the range of tasks over a wide, compared to back propagation networks. That is, the feed forward networks are imposed constraints: all the signals in the network extend only from input to output, but not vice versa;-layered network is supposed to mesh, modeling of dynamic processes in such networks is possible only by artificial methods as feed forward networks have no internal state (the values of output neurons depend only on the input vector and do not vary over time, if the input is unchanged). These limitations narrow the possibilities of the model. In addition, a network with feedback can act as an associative memory. This means that the vector filed as input, the network will be created at the output of one of the most "similar" (in some sense chosen) to the previously memorized vector. This method of sampling data is addressed by content, as opposed to addressing by number of memory cells, taken in a computer-von Neumann type. Memory addressing is a very promising for the creation of artificial intelligence systems.

$$\mathrm{NET}_j = \sum_{i \neq j} w_{ij} \mathrm{Y}_i + \mathrm{X}_j \quad (7)$$

Y, = 1 if $\mathrm{NET}_j > T_j$,

Y = 0 if $\mathrm{NET}_j < T_j$,

Y does not change if $\mathrm{NET}_j = T_j$,

The disadvantage of the Hopfield networks is their tendency to stabilize in a local rather than global minimum of the energy function. This difficulty can be overcome largely by a class of networks known as the Boltzmann machine, in which changes in the states of neurons caused by statistical rather than deterministic laws. At a fixed temperature distribution of the energy system is determined by a probabilistic Boltzmann factor exp (-E/kT), where E - energy of the system; k - Boltzmann constant, T - temperature. From here you can see that there is a finite probability that the system has a high energy, even at low temperatures. Similarly, a small but calculated the probability that a kettle of water on the fire freezes before the boil.

The statistical distribution of energy allows the system out of local minima of energy. At the same time, the probability of high-energy states decreases rapidly with decrease in temperature. Consequently, at low temperatures there is a strong tendency to take a low-energy state.

To implement the resulting system introduces the probability of weight change as a function of the amount by which the output neuron Y is greater than its threshold. Let

$$E_k = \mathrm{NET}_k - \theta_k, \quad (8)$$

when NET_k — output NET neuron k; θ — the threshold of neuron k, and

$$p_k = \frac{1}{1 + \exp(-\delta E_k / T)} . \quad (9)$$

Under the operation of artificial temperature T is attributed to the importance of the neurons are established in the initial state determined by the input vector, and the network an opportunity to seek the minimum energy in accordance with the following procedure:

1. Attributed to the state of each neuron with a probability value of the unit and p_k, and with probability $1 - p_k$ - zero.

2. Gradually reduce the temperature of the artificial and repeat step 1 until equilibrium is reached.

The stability of such a network can be proved using an elegant mathematical method. Assume that some function, which always decreases when the network status. In the end, this function must reach a minimum and to stop the change, thus ensuring the stability of the network. Such a function is called a Lyapunov function for the considered networks with feedback can be introduced as follows:

$$E = -\frac{1}{2} \sum_i \sum_j w_{ij} \mathrm{Y}_i \mathrm{Y}_j - \sum_j I_j \mathrm{Y}_j + \sum_j T_j \mathrm{Y}_j \quad (10)$$

where E - energy of the artificial network; w_{ij} - the weight of the output neuron i to input neuron j; Y_j - the output of neuron j; I_j - external input of neuron j; T_j - the threshold of neuron j.

The change in energy E, due to changes in state j-neuron is

$$\delta E = \left[\sum_{i \neq j} (w_{ij} Y_i) + I_j - T_j \right] \delta Y_j = -[\text{NET}_j - T_j] \delta Y_j \qquad (11)$$

where δY_j - change of the j-th output neuron.

Learning procedure for the resulting network consists of the following steps:

1. Calculate the fixed probability.

a) give the input and output neurons values of the training vectors;

b) provide the network to find a balance;

c) write the output values for all neurons;

d) repeating steps a through in all training vectors;

e) calculate the probability P_{ij}^{+}, that is, the whole set of training vectors to calculate the probability that the values of both neurons are equal to unity.

2. Calculate the loose credibility.

a) provide a network to a "free traffic" without fixing the inputs or outputs, starting from a random state;

b) repeat step 2 a lot of times, recording values of all neurons;

c) calculate the probability P_{ij}^{-}, ie, the probability that the values of both neurons are equal to unity.

3. Adjust the network weights as follows:

$$\delta w_{ij} = \eta (P_{ij}^{+} - P_{ij}^{-}), \qquad (12)$$

where δw_{ij} - the change in weight w_{ij}, η - learning rate coefficient.

To begin working with an artificial neural network Hopfield - it must be trained on the training set. For a given flight training flying object, which is planned for the complex. The trajectories of these flights will be our training sample. These trajectories are described by the set of vectors $X_j = (x_1, \ldots, x_i, \ldots, x_n)$, where j ∈ [1, m], m - number of points taken to describe the flight. The vector X_j is a set of parameters characterizing the position x_i aircraft flying at a certain point.

Training the network is as follows: the input is set to the output vector X, respectively, the set of vectors Y. The vector Y - output vector, the structure is similar to the vector X. Given that the vector X is actually the result of a training flight, the X and Y are the same. Therefore, the entire learning process is reduced simply to achieve a "minimum" weight w_j^i, where j - number of neurons, i - number of X

After training the neural network is ready to work, whose aim is to recognize the input noisy signals and give the result. Figure 5 shows the block diagram shown of the Hopfield ANN.

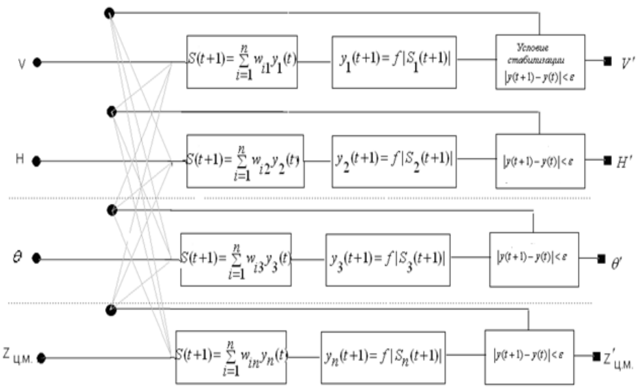


Figure 5 - Block diagram of the Hopfield ANN

The input vector X alternately serve the form X = (V, H, θ, ...,$Z_{ц.м.}$). For each parameter, we calculate the neuron state S (t +1) and the output Y (t +1), we check the stability of output parameters to look at the difference between the two outputs of the last two iterations, if the output is not stable, then calculate the new state of the neuron S (t +1) and repeat the cycle, taking into account other parameters. When all outputs stabilize we send them to the output, i.e. obtain the vector Y(V', H', θ',..., $Z'_{ц.м.}$).

In view of the limited Hopfield neural network on the amount of memorable images, it was decided to apply the algorithm forgetting. The possibility of forgetting unnecessary, superfluous information is one of the remarkable properties of biological memory. The idea of application of this property to an artificial neural network Hopfield "surprise" is simple: when memorizing the training set of images together with them to remember and false images. They are something, and should "forget".

The corresponding algorithms are called algorithms forgetting. Their essence is as follows.

In the first phase is training the network according to the standard Hebb rule. The memory is filled with real images and a lot of false information. In the next phase (phase forgetting) network presented some (random) image of $\lambda^{(0)}$ network evolves from state $\lambda^{(0)}$ to some state $\lambda^{(f)}$, which is a high-volume training sample more all is false. Now the matrix of relationships can be corrected in order to reduce the depth of the minimum energy corresponding to this false state:

$$S_{ij}(t+1) = S_{ij}(t) - \varepsilon * \lambda_i^{(f)} * \lambda_j^{(f)} \quad (13)$$

As the extent of forgetting ε is chosen for a small number, which guarantees a slight deterioration of memory useful, if the condition $\lambda^{(f)}$does not prove false. After a few "sessions forgetting" the network properties are improved. Thus, we obtain the parameters that will be the benchmark for the aircraft at any given time. And if you need to take control of the aircraft complex, these parameters will be passed to block the decision-making, where the signals will be formulated, which will be served on the steering mechanism of the aircraft to achieve the standard parameters. So plan to exit from a dangerous situation.

This system is in exploring the concept of virtual testing, and individual items.

3. Summary

The paper presents the highlights of the implementation of these tasks are shown binding elements. In addition, has been studied and confirmed the possibility of the system (based on operational analysis data external trajectory measurements, evaluation of flight characteristics and parameters of the status and functioning of the system) to predict the development of dangerous and critical situations, and provide information about the pilot approached him. Verified estimate of the probability aircraft control to means of the developed complex. Presented and analyzed the preliminary results.

Project completion and implementation of safety for civilian control of the court will provide real-time systems and components airplane pilot's actions in moments of approaching hazards, as well as its information support in order to prevent accidents, loss of aircraft and crew. Through the application of information technology based on neural network algorithms that can predict the possible critical flight situation, the pilot will have a large

References

[1] Kucheryavyi A.A. «Board information systems». Ulyanovsk: Univ of Ulyanovsk State University, 2004

[2] Soldatkin V.M. «Methods and tools for building on-board information management systems to ensure flight safety». Kazan Univ Kazan State. those. University. 2004

[3] Williams-Hayes P.S. «Flight Test Implementation of a Second Generation Intelligent Flight Control System». NASA.TM-2005-213669 , 2005

Copyright Statement