AN ALGORITHM FOR SOLVING THE PROBLEM
OF FORECASTING

Victor Sineglazov[1], Elena Chumachenko[2], Vladyslav Gorbatyuk[3]

National Technical University of Ukraine 'KPI', Peremogy Ave. 37, 03056 Kiev, Ukraine
E-mail: [3]lobach21@mail.ru (corresponding author)

Victor SINEGLAZOV, *Prof, Dr*
Date and place of birth: 1950, Odessa, Ukraine.
Education: National Technical University of Ukraine 'KPI'.
Affiliation and functions: Doctor of Technical Sciences, professor, has worked at National Aviation University since 1989, director of the Institute of Electronics and Control Systems at the National Aviation University, head of the Department of Aviation Computer-Integrated Systems since 1998, active member of the International Academy of Navigation and Control and of the New York Academy of Sciences.
Research interests: automated control systems and information processing.
Publications: over 485 publications.

Elena CHUMACHENKO, *Assoc. Prof*
Date and place of birth: 1950, Kutaisi, Georgia.
Education: Georgian Polytechnic Institute.
Affiliation and functions: PhD., associate professor, has been working at National Technical University of Ukraine 'KPI' since 1976.
Research interests: modelling and management of flexible computer systems.
Publications: over 35 publications.
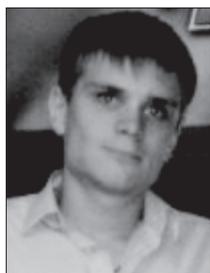
Vladyslav GORBATYUK
Date and place of birth: 1992, Ilyichyivsk, Ukraine.
Education: National Technical University of Ukraine 'KPI'.
Affiliation and functions: 4th year student at National Technical University of Ukraine 'KPI'.
Research interests: artificial neural networks, forecasting methods.
Publications: two articles.

Abstract. In this article, the main forecasting methods are considered. A new algorithm based on the group method of data handling and artificial neural networks is presented. The algorithm was tested on real data and showed better results than neural networks. This shows its suitability for further use in forecasting aviation tasks.

Keywords: artificial neural network, forecasting method, algorithm, mean square prediction error, group method of data handling, training set.

1. Introduction

Forecasting has always been and will always be one of the most interesting themes to mankind because knowledge of the future is perhaps one of man's greatest desires, but we should understand the risk of choosing inappropriate methods of forecasting, since incorrect predictions can lead to making the wrong decisions.

Demand prediction is one of the most crucial issues of inventory management. Forecasts, which form the basis for the planning of inventory levels, are probably the biggest challenge in the repair and overhaul industry, and the one common problem facing airlines throughout the world is the need to know the short-term demand forecast for aircraft with the highest possible degree of accuracy. The high cost of modern aircraft and the expense of repairable spares such as aircraft engines and avionics constitute a large part of the total investment of many airline operators. The production of an insufficient amount of aircraft can lead to excessive downtime costs, while overproduction will result in downtime.

Forecasting techniques in aviation have grown more sophisticated over the years and are widely used in aviation nowadays. Owing to the sporadic nature of demand for aircraft, airline operators perceive difficulties in prediction and are still looking for superior forecasting methods.

The purpose of this paper is to develop a new forecasting method that takes into account and uses the increased power of modern computing. It is suggested that our algorithm may further be applicable to regional aviation facilities and other industrial sectors that have demand patterns similar to those of airlines.

2. The mathematical formulation of the problem of forecasting

Let us have n discrete samples $\{x_1, x_2, ..., x_n\}$ at successive time points $t_1, t_2, ..., t_n$. Then the problem of prediction (Fig. 1) consists of predicting the value x_{n+k} at some future point of time t_{n+k}, where k is the duration of the forecast:

$$x_{n+k} = F(x_1, x_2, ..., x_n), \qquad (1)$$

where F is some unknown function.
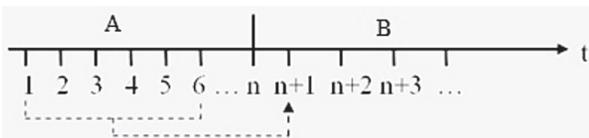
Fig. 1. Graphical illustration of problem statement forecasting: A – known values, B – forecast period

3. Review of existing methods of forecasting

Here are the basic methods of forecasting.

The method of moving average (Alesinskaya 2002). This method is based on a simple model that assumes that the current value of a number

$$y_t, (t = 1...n), \qquad (2)$$

of the raw

$$\{y_1, y_2, ..., y_n\}, \qquad (3)$$

is the average sum of a number of previous values and some random component.

Weighted moving average method (Alesinskaya 2002). The next step in the modification of the model is the assumption that the more recent values reflect the situation more accurately. Then, each value is assigned greater weight the later it was added.

Group method of data handling (GMDH) (Ivakhnenko 1968). The GMDH is a set of forecasting algorithms that is based on splitting the original data into two sets (training and testing), and the use of some kind of base functions, the parameters of which are derived from the training set. Verification of how well they model a given process is performed on the test set.

An artificial neural network (ANN) (Mak-Kalloc 1956). The main element of a neural network is a formal (artificial) neuron. It is a mathematical model of biological neural cells.

An ANN is a system of connected and interacting artificial neurons.

An ANN is not programmed in the usual sense of the word: they can be trained. During training, the neural network is able to detect complex relationships between input and output data and perform synthesis.

The ability of neural networks to forecast comes directly from their ability to generalise and find the hidden relationships between input and output data. After training, the network is able to predict the future value of a certain sequence on the basis of several previous values and/or any current factors.

The main advantage of an ANN over other methods of forecasting is that the network equally well predicts the processes whose regular components have any distribution law, whereas most other methods are best suited for processes that have a regular component that belongs to a specific class (obviously, the method of polynomial smoothing is best suited for processes with a polynomial regular component, the method of smoothing by Fourier series is best suited for processes with a periodic regular component, etc.). Another important advantage of neural networks is the ability to learn.

The application of artificial neural networks (namely time-lagged feed-forward neural networks) for prediction of passenger traffic flows was described by T. O. Blinova, and they showed reasonably good results (Blinova 2007). However, as will be shown in the application results of the proposed algorithm, its performance is far better than the performance of neural networks on their own.

Another application of time-lagged feed-forward neural networks for aircraft engine overhaul demand forecasting was mentioned by P. Kozik and J. Sęp and again, neural networks showed good performance, which justifies their use for forecasting tasks (Kozik, Sęp 2012).

In this situation, a forecasting algorithm based on two common methods, ANN and GMDH, is offered.

4. An algorithm for solving the problem of forecasting

The proposed algorithm combines the advantages of both ANN and GMDH: a gradual increase in the complexity of the model (GMDH) and the ability of ANN to learn. It consists of the following steps:

1. Pre-processing of source data, including getting rid of outliers, data normalisation, etc. (Amir, Samir 1999; Jerome *et al.* 1994; Klevecka, Lelis 2008; Mohsen, Yazdan 2007). This stage is often more important than the modelling stage. Also you should take into account the specific characteristics of the forecasted process, such as seasonality of values in the case of predicting various atmospheric parameters, the trend component in the majority of financial processes, etc.

Study shows the importance of getting rid of unwanted outliers in the source data when using artificial neural networks (Jerome *et al.* 1994). It is important to determine whether the specific value is accidental and therefore an unwanted outlier or whether it is an informative outlier. To do this, you should consider the process you are forecasting. For example, if you want to build a model of energy consumption of some region depending on the day of the week, then it should be noted that energy consumption on weekends will be dramatically different from the consumption on weekdays, and consequently the value of consumption over the weekend will not constitute unwanted outliers. In this case, during the training of the network, it would be reasonable to include a binary value as an additional input showing whether this day is on the weekend or not.

If it is known that the process does not include informative outliers, there are several basic algorithms of getting rid of accidental outliers.

– The simplest algorithm is based on the characteristics of a stochastic variable, according to which an outlier is a value that deviates from the aver-

age value by an amount greater than 2 ... 3 mean square deviation σ^2.

– The Tukey 53H algorithm consists of the construction of a smoothed sequence using a median filter and a moving average filter, and after its application all original values that deviate from the smoothed sequence more than a pre-assigned threshold k will be considered outliers (Klevecka, Lelis 2008).

2. With the use of data windowing, two matrices are formed from the source data

$$X = \left\{ x_n, n = 1 \dots N \right\}, \tag{4}$$

where N is the number of samples obtained from the previous stage of processing:

$$X^{(0)} = \begin{bmatrix} x_1 & x_2 & \cdots & x_k \\ x_{k+2} & x_{k+3} & \cdots & x_{2*k+1} \\ \cdots & \cdots & \cdots & \cdots \\ x_{N-k-1} & x_{N-k} & \cdots & x_{N-1} \end{bmatrix}, y = \begin{bmatrix} x_{k+1} \\ x_{2*k+2} \\ \cdots \\ x_N \end{bmatrix}, \tag{5}$$

where k is the size of the window. With the help of these matrices, networks will be trained. Each row vector

$$\overline{x}_n = \left\{ x_{n1}, x_{n2}, \dots, x_{nk} \right\}, n = 1 \dots m, \tag{6}$$

where m is a number of rows, of matrix $X^{(0)}$ and the corresponding value y_n are an independent sample, and a column vector

$$\vec{x}_z = \left\{ x_{z1}, x_{z2}, \dots, x_{zm} \right\}, z = 1 \dots k, \tag{7}$$

is a separate variable.

3. The resulting samples are divided according to some ratio (usually about 0.7:0.3 m) into training and testing sets.

4. The type of base function is determined by what variables they depend on, for example

$$f_l = f\left(x_i, x_j \right), \tag{8}$$

or

$$f_l = f\left(x_i, x_j, x_i * x_j, x_i^2, x_j^2 \right), \tag{9}$$

where $l = 1 \dots C_k^2$ is the number of possibilities to choose two variables from k possible (in the general case $l = 1 \dots C_k^O$, where O is the number of variables taken as arguments of support functions). In contrast to the GMDH, where the concrete form of functions is also determined (for example $f_l = a_0 + a_1 * x_i + a_2 * x_j$), in this algorithm only variables on which the base functions will depend are selected (the powers of the variables and their covariance can be represented as additional variables), while the determination of the dependence

of function values on variables will be performed by the neural network.

5. C_k^O simple multilayer perceptrons (MP) are constructed (O is the number of variables taken into account in support functions, for the above example o = 2) with one output, one hidden layer with a small amount of neurons (about three) in it, and the right amount of inputs (for support functions with the form $f_l = f(x_i, x_j)$, a network must have two inputs, for $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$, it must have five inputs, etc.).

6. Each perceptron is matched with a specific base function; namely the variables to be fed to the inputs of the network are selected (for example, for support functions of form $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$, one MP will work with variables $x_1, x_2, x_1 * x_2, x_1^2, x_2^2$ and the other one will work with variables $x_1, x_k, x_1 * x_k, x_1^2, x_k^2$), and are trained using only samples from the training set.

7. In this step, the initial data for the next iteration of the algorithm should be composed. For this, you should determine the mean square error (MSE) of each MP on the test set, select k best networks (you can also select a smaller amount, but add to them original variables that were inputs to networks with a small MSE), and then create a new matrix:

$$X^{(1)} = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1k} \\ h_{21} & h_{22} & \dots & h_{2k} \\ \dots & \dots & \dots & \dots \\ h_{m1} & h_{m2} & \dots & h_{mk} \end{bmatrix}, \tag{10}$$

where h_{ij} is the output value of the jth neural network when it is given the inputs of the ith sample, $i = 1 \dots m, j = 1 \dots k$ (or the original variable).

8. The next iteration is performed, but as the source data a matrix, $X^{(1)}$, is taken. Iterations are performed until the MSE value of networks on the test set decreases or until we reach the desired MSE.

9. During an each iteration of the algorithm, the weights and the structures of networks (and/or the original variables) that were selected to make the source matrix for the next iteration should be remembered. After reaching the required MSE (or reaching the iteration after which MSE starts increasing), we should stop performing the iterations and select the only network that will forecast the further values. For further forecasting of some value x_c using the obtained results we should:

- Create source sample $inp = \{x_{\tilde{n}-k-1}, x_{\tilde{n}-k}, \dots, x_{\tilde{n}-1}\}$;
- Use this sample as the inputs for the MPs that were used to obtain the source matrix of the second iteration, and using the output values obtained (and/or the original variables), make a new input sample for the second iteration;

- Repeat the previous step until the input sample for the iteration at which the algorithm was stopped is created, and then use it as the input for the selected MP, the output of which will be a forecast of the value x_c.

5. Advantages and disadvantages of the algorithm

The advantage of this algorithm compared with the usual GMDH is that explicit specification of the basic functions is not required; the dependence will be found by the neural networks, which are known to perform this task very well.

This algorithm is also devoid of some drawbacks of ANNs: first, during the construction of the network, its optimal difficulty is not known beforehand, and second, overly simple neural networks are usually prone to underfitting, while overly complex networks are prone to overfitting. The proposed algorithm uses simple networks at each iteration, but due to cascade complication it is able to forecast very complex processes.

The main drawback of the proposed method is its resource usage, caused by building some number of neural networks at each iteration, so it can take a lot of time. However, with the use of numerically optimised algorithms of neural network training and general algorithm optimisation, it is possible to reach relatively fast performance.

6. Application results of the proposed algorithm

To test the proposed algorithm, a public set of aircraft sales (total number of aircraft sold in the USA per year) from 1947 to 2011 was used (Accident Database … 2012). The data was pre-processed using the Tukey53H algorithm.

To realise the particular algorithm, the following parameters were used:

- The size of a sliding window for obtaining matrices of initial samples: $k = 5$;
- Ratio training set size/testing set size: $0.7m / 0.3\,m$, where m is the number of initial samples. For the construction of the training set, the first 0.7 m samples were used;
- Basic function form: $f_l = f(x_i, x_j, x_i * x_j, x_i^2, x_j^2)$, $l = 1 \dots C_5^2$, $i = 1 \dots 5, j = 1 \dots 5, i \neq j$;
- Structure of neural network: single output neuron, one hidden layer with three neurons, five input neurons;
- MSE for selection of networks was calculated on all initial samples, which means on testing and training sets;
- For the construction of the source matrix for the next iteration, the outputs of three networks with the minimal MSE on training and testing sets were taken, along with two initial variables

that were the inputs for the network that showed the smallest MSE.

For comparison of the results of the application of the algorithm, a multilayer perceptron with single output neuron and one hidden layer having eight neurons and ten inputs was also built. MP, as well as the proposed algorithm, was trained only on the training set. The Levenberg-Marquardt algorithm was used as a training algorithm. Since it is known that ANNs are prone to getting stuck in local minimums of the network's cost function, about ten MPs were trained, and the perceptron with the minimal MSE was chosen.

Comparison of the results obtained with the use of ANN and the proposed algorithm is shown in figure 2.
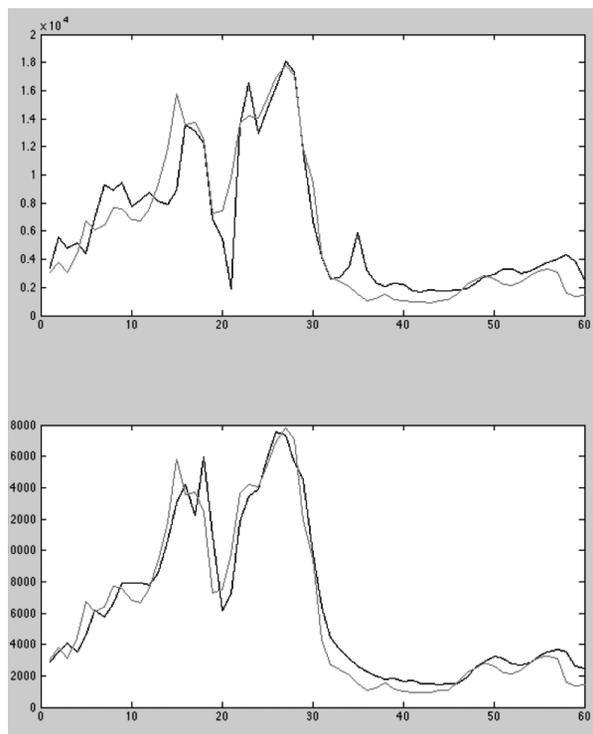
Fig. 2. Comparison of the prediction results of ANN (on top) and the proposed algorithm (bottom) (with data pre-processed using the Tukey53H algorithm). ———– actual data, ———– forecast

The MSE values: ANN – 0.0613, algorithm – 0.0255.

From the figure, it can be noticed that the trained perceptron predicts the values of the testing set worse (the very last values), while the proposed algorithm forecasts the values of both training and testing sets equally well, indicating the more accurate model of the process. The algorithm also more correctly predicted extreme values, which is very important in demand forecasting.

7. Conclusions

The proposed algorithm has shown significantly better results in comparison with the artificial neural networks, a sign of its suitability for further usage in forecasting tasks.

A possible way to further improve the algorithm is to consider using different neural network structures (not only inputs) for each neural node. This will require determining how exactly to change the structure for each node, however.

References

Alesinskaya, T. V. 2002. *Metodi skolziaschego srednego i eksponencialnogo sglagivaniya*. Uchebnoe posobie po resheniu zadach po kursu 'Ekonomiko-matematicheskie metodi i modeli'. Taganrog: TRTU. 153 p. (in Russian).

Amir, F. A.; Samir, I. S. 1999. A comparison between neural-network forecasting techniques – case study: river flow forecasting, *IEEE Transactions on Neural Networks* 10(2): 402–409. http://dx.doi.org/10.1109/72.750569

Blinova, T. O. 2007. Analysis of possibility of using neural network to forecast passenger traffic flows in Russia, *Aviation* 11(1): 28–34.

Ivakhnenko, A. G. 1968. Metod gruppovogo urakhuvannya argumentiv – konkurent metodu stokhastichnoi approksimacii, *Avtomatika* (3): 58–72 (in Ukrainian).

Jerome, T. C.; Douglas, R. M.; Atlas, L. E. 1994. Recurrent neural networks and robust time series prediction, *IEEE Transactions on Neural Networks* 5(2): 240–254. http://dx.doi.org/10.1109/72.279188

Klevecka, I.; Lelis, J. 2008. Pre-processing of input data of neural networks: the case of forecasting telecommunication network traffic, *Telektronik* 3–4: 168–178.

Kozik, P.; Sęp, J. 2012. Aircraft engine overhaul demand forecasting using ANN, *Management and Production Engineering Review* 3(2): 21–26.

Mak-Kalloc, P. 1956. Logicheskoe ischislenie idei, otnosyashikhsya k nervnoi aktivnosti, *Avtomaty*, 363–384. (in Russian).

Mohsen, H.; Yazdan, S. 2007. Artificial neural network approach for short term load forecasting for Illam region, *Engineering and Technology* 28: 280–284.

U.S. General Aviation Aircraft Shipments and Sales [online] 2012. [Cited August 2012]. Available from Internet: http://www.bga-aeroweb.com/database/Data3/US-General-Aviation-Aircraft-Sales-and-Shipments.xls