ISSN 2278-3091
Volume 8, No.1.2, 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse0181.22019.pdf
https://doi.org/10.30534/ijatcse/2019/0181.22019

The Method of Improving the Efficiency of Routes Selection in Networks of Connection with the Possibility of Self-Organization

Nina Kuchuk[1], Amin Salih Mohammed[2], Andrii Shyshatskyi[3], Oleksii Nalapko[4]
[1]Department of Theoretical and Applied Systems Engineering, V. N. Karazin Kharkiv National University, Kharkiv, Ukraine, nina_kuchuk@ukr.net
[2] Assistant Professor, Department of Computer Engineering, Lebanese French University, Erbil, KR-Iraq kakshar@lfu.edu.krd
[3, 4]Central Research Institute of Weapons and Military Equipment of Armed Forces of Ukraine, Kyiv, Ukraine, ierikon12@gmail.com

ABSTRACT

One of the directions improving the efficiency of communication networks with the ability to self-organization is the choice of rational route information transmission from sender to recipient. In the course of the analysis it was established that the existing scientific approaches for forecasting the state of the routes have high computational complexity, which prevents them from predicting the state of the transmission routes in real time. Therefore, the authors of this research conducted a methodology for forecasting the time overload of data transmission routes in mobile radio networks of radio networks with the possibility of self-organization. As the basic mathematical device, the neural network of Elman was used based on the calculation of the network neuron potential. The essence of the proposed method is to decide on the search for new routes based on the predicted time of overloading data transmission routes in mobile radio networks with the possibility of self-organization. The proposed method allows to predict the time of overloading of data transmission routes in mobile radio networks with the possibility of self-organization by reducing the computational complexity of the neural network and application of the algorithm for training the neural network. It is advisable to use the proposed methodology in radio communication with programmable architecture by developing appropriate software, which will increase the efficiency of choosing a route of data transmission in radio communication networks with the ability to self-organize.

Key words: networks with the ability to self-organize, routing, neural networks, mobile nodes.

1. INTRODUCTION

The peculiarities of the functioning of MANET (Mobile Ad-Hoc Network) class networks including the ability to self-organization, dynamic topology, a large flow of data, application in uncertainty, mobility of nodes. Also, to the peculiarities of the operation of the mobile radio link (MRL) node runs on a battery, the limited capacity of which affects the functioning time of both the mobile node and the network as a whole [1, 2].

In the MRL MANET class, the transmission of information between the receiver-transmitting components of the network can be carried out either directly or through relaying through intermediate nodes. In the case of a large amount of information over communication networks, one of the main conditions for the effective functioning of the MRL is the continuous transmission of information in the absence of overloads of nodes and communication channels MRL. In turn, the directions of prevention of overloads are the coordination of transmission and reception rates, the formation and streamlining of traffic, management of the connection between the sender and the destination, management of data transmission routes. In general, the implementation of the above-mentioned directions for preventing overloads may be a task for forecasting data flow overload. Therefore, the purpose of this publication is to develop a method for predicting the time overload of data transmission routes in the MANET class mobile radio networks.

2. LITERATURE SURVEY

In the MRL MANET class, which are characterized by high dynamics of change in the topology, the presence of uncertainty in the network, the processing of a large number of heterogeneous data, the transfer of information between the receiver-transmitting components of the network can be carried out either directly or through relaying through intermediate nodes. However, in any case, the efficient transmission of information between the subscribers of the MRL can occur only in the absence of overloading the routes of data transmission in the MRL. The presence of a network overload, loss of communication channels or a break in the routes of information transmission in the process of transmitting information, will lead to the impossibility of maintaining communication between nodes and units. The presence of this factor can directly affect the quality of information transmission, the necessity of searching for a new routes information transmission [3–5].

An analysis of existing methods for preventing overload [6] indicates that in these methods, the main functional features are: orientation for the use in stationary networks, route construction, route maintenance, route management, message queue management, inability to make control decisions in the condition of uncertainty, necessity of work with significant volumes of service traffic.

The main drawbacks of these methods include early termination of the route between the destinations, packet delays, packet loss, the need for administrator intervention, the absence of internal monitoring of the operation, etc. [6].
In [6, 7], it is noted that effective devices of controlling overload can be monitoring the traffic, which is carried out in order to detect overloads and factors that affect the network overload. The main tasks of the implementation of this tool are: control of the capacity of communication channels, detecting overloads, detecting the failure of the software and hardware components, detecting user activity and events, detecting network vulnerabilities and violations in settings and forecasting the network.

The presence of functional features of methods for preventing overload and the disadvantages of their application stipulate a list of requirements for methods of preventing overload, in case of their application in the MRL, namely: decentralization of management; forecasting of the network overload time; decision making in conditions of uncertainty; reduction of the time for making a management decision.

Proceeding from the indicated, actual direction of research aimed at developing new and improving the existing intelligent methods of forecasting overload, it may be decision-making on prediction of overload time in MRL.

3. SETTING UP THE TASK OF SCIENTIFIC RESEARCH

Due to the impossibility of collecting real-time information about the state of the MRL, we will consider the process of routing data flows in the information direction *a* – *b*, which consists of the final nodes *a* and *b* (respectively, the sender and the addressee), as well as the set of nodes that form transmission channels between *a* and *b*. Suppose there is a route m between nodes *a* and *b*, in which the total number of nodes is *k*. The nodes can change transmitter power $p_i(t) \le p_{i\max}$. Parameters of nodes $e_i(t)$, $i = \overline{1, N}$; $T_i(t)$ are the time of life of the *i*-th node. Radio channel parameters $s_i(t) \le s_{i\max}(t)$. Type of information is the ξ = 1–3 (language, video, data transmission); number of recipients at each session $|b| = 1$ (single-player transfer). Radio interconnection between the nodes of the network is supported by the channel level protocol; Signal strength and signal / noise ratio are considered unchanged. Time of existence of the current route $T_m(t)$ is determined by the minimum time "life" of the node $T_i(t)$ on the route m: $T_m(t) = \min(T_1(t), T_2(t), ..., T_i(t))$, $i = \overline{1, k}$. Network status parameters: x_1 is the traffic type, x_2 is the volume of information, x_3 is the number of recipients, x_4 is the size queues, x_5 is the time of existence of the route, x_6 is the rate of change of the size of the queue, x_7 is the packet loss ratio, x_8 is the packet transmission delay in the network, x_9 is the channel capacity[16].

It is necessary: to predict the time overload of data transfer routes in the MRL.

The essence of the method is to decide on the search for new routes based on the predicted time of the transfer routes, in order to meet the requirements of user optimization:

$$U(t) = \arg \min_{U(t) \in \Omega} V(X, U(t)). \qquad (1)$$

4. NEURAL NETWORK BUILDING ARCHITECTURE

While choosing the architecture of the developed method of forecasting the time of overload in the MRL should take into account that prediction is a special case of the problem of regression, that is, the dependence of a dependent variable from independent in the given conditions, then the solution to this problem may be the use of neural networks (NN), namely: multilayered perceptron, radial-base network, generalized regression network, Volter network and Elman's network [8, 9].

The analysis of the use of NN in solving tasks of prediction of events indicates the expediency of applying the calculation of time series based on which the Elman neural network, which is one of the types of recurrent network, will be laid. Elman's network consists of a multi-layered perceptron with feedback. This feature allows you to take into account the previous actions and accumulate information to support the adoption of management decisions based on the forecasting of the time series. The forecasting of the time series is reduced to the problem of interpolation (the intermediate values definition) of the function of many variables and the solution of the approximation problem (bringing to a simplified form) a multidimensional function that has an undeniable effect on the prediction quality [8, 9].

The network of Elman consists of three layers: entrance, hidden and outgoing. In this case, the hidden layer has a feedback on itself, as a result of which the training of the neural network, analysis of the events that have occurred and as a consequence of prediction of the future. Figure 1 shows the scheme of the neural network of Elman [10, 11,12,13,14].

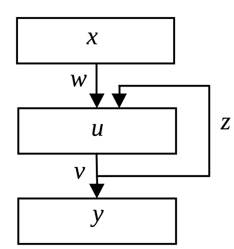

Figure. 1: Elman`s neural network scheme

In contrast to the usual network of direct propagation, the input image of the recurrent network is not one vector, and the sequence of vectors $\{x(1), x(2), ..., x(n)\}$ of the input image is fed to the input in a given order, with the new state of the hidden layer depending on its previous states. The Elman network may be described by the following relationships:

$$u(t) = f(w \cdot x(t) + z \cdot u(t-1) + b_h), \quad (2)$$

$$y(t) = g(v \cdot u(t) + b_y), \quad (3)$$

where $x(t)$ is the input vector t number ; $u(t)$ is the condition of the hidden layer for the entrance $x(t), (u(0) = 0)$; $y(t)$ is the output of the network for the entrance $x(t)$; z is the weight matrix of the distribution layer; v is the weight (square) matrix feedback of the hidden layer; b_h is the vector of shifts of the hidden layer; V is the weight matrix of the output layer; b_y is the vector of shifts of the output layer; f is the activation function of the hidden layer; g is the activation function of the output layer.

To explore the Elman's network gradient methods [10, 11] are used, which results in the neural network were calculated by the method of back propagation with the deployment of the network in time [12].

The architecture of the modified recurrent Elman neural network for forecasting the time of route overload is shown in Fig. 2. Using the network assumes that the prediction process is simulated by the output signal of some nonlinear dynamic system, which depends on a plurality of factors, including the previous condition of the system [15]. A layer of feedback has been introduced in the network. This layer receives signals from the output of the hidden layer and delivers them to the input layer z through the delay elements, thus storing the processed information from previous network cycles [8–10].

Considering the prediction of overloading routes in the network, standard neurons with activation functions, delay elements z , and phaseization blocks are typically used to convert the inputs of the order and nominal variables characterizing the network's influence into a quantitative form.

Fig. 2 shows Elman network for predicting the time that overload routes with multiple inputs, where the number of neurons in the input layer m and a hidden layer n and one output block. Let it be x_{it} $(i = 1, 2, ..., m)$ denote the set of input vectors of the neurons at the time t , y_{t+1} denotes the output of a network at a time point $t+1$, u_{jt} $(j = 1, 2, ..., n)$ denote the conclusion of the latent layer neurons over time t and z_{jt} $(j = 1, 2, ..., n)$ denote the recurrence layer neurons, where w_{ij} is the weight that connects the knot i in the input layer of the neurons to the knot j hidden layer; c_j, v_j is the scales connecting the knot j in the neurons of a hidden layer with a knot in the recurrence layer [10].

Inputs of the hidden layer neurons:

$$NET_{ji}(k) = \sum_{i-1}^{n} w_{ij} x_{it}(k-1) + \sum_{j-1}^{m} c_{ij} z_{it}(k), \quad (4)$$

where $z_{ji}(k) = u_{jt}(k-1)$, $i = 1, 2, ..., n$, $j = 1, 2, ..., m$.

Outputs of the hidden layer neurons:

$$u_{ji}(k) = f_H\left(\sum_{i-1}^{n} w_{ij} x_{it}(k) + \sum_{j-1}^{m} c_{ij} z_{it}(k)\right), \quad (5)$$

where the sigmoidal function in the hidden layer is selected as an activation function: $f_H(x) = 1/(1+e^{-x})$ [10].

Elman's network algorithm with stochastic efficiency of time. The back propagation algorithm is a controlled training algorithm that minimizes the global error E using the gradient descent method [10–12]. For the model of the stochastic efficiency of the Elman's network time, we assume that the received output error $\varepsilon_{e_m} = d_{t_n} - y_{t_n}$ and sample error n is defined as:

$$E(t_n) = 0{,}5\phi(t_n)\left(d_{t_n} - y_{t_n}\right)^2, \quad (6)$$

where t_n is the response time n $(n = 1, 2, ..., N)$, d_{t_n} is the actual value, y_{t_n} is the entrance at the time t_n , $\phi(t_n)$ is an effective function of stochastic time. Effective time data function is considered as a function of the time variable:

$$\phi(t_n) = \frac{1}{\beta}\exp\left\{\int_{t_0}^{t_n} \mu(t)dt + \int_{t_0}^{t_n} \sigma(t)dB(t)\right\}. \quad (7)$$

Network data error is defined as:

$$E = \frac{1}{N}\sum_{n-1}^{N} E(t_n) = \frac{1}{2N}\sum_{n-1}^{N} \phi(t_n) \cdot \left(d_{t_n} - y_{t_n}\right)^2. \quad (8)$$

The main task of the learning algorithm is to minimize the value of the state function of the network E until it reaches a given minimum value ξ by re-training. At each repetition, the conclusion is calculated and a global error is obtained. The gradient of the network state function is determined $\Delta E = \partial E / \partial W$. For nodes of weight in the input layer, the gradient of connecting weights w_{ij} is given by the formula:

$$\Delta w_{ij} = -\eta \frac{\partial E(t_n)}{\partial \omega_{ij}} = \eta \varepsilon_{t_n} v_j \phi(t_n) f'_H(NET_{jt_n}) x_{it_n}, \quad (9)$$

for the knot weights in the recurrence layer is given by the formula:

$$\Delta c_j = -\eta \frac{\partial E(t_n)}{\partial c_{ij}} = \eta \varepsilon_{t_n} v_j \phi(t_n) f'_H(NET_{jt_n}) z_{it_n}, \quad (10)$$

for the knot of weight in a hidden layer is given by the formula:

$$\Delta v_j = -\eta \frac{\partial E(t_n)}{\partial v_j} = \eta \varepsilon_{t_n} v_j \phi(t_n) f'_H(NET_{jt_n}), \quad (11)$$

where η is the learning speed $f'_H(NET_{jt_n})$, is a derivative of the activation function.

Updated rules for the weights w_{ij} , c_j and v_j are adjusted as:

$$w_{ij}^{k+1} = w_{ij}^{k} + \Delta w_{ij}^{k}, \qquad (12)$$

$$c_{j}^{k+1} = c_{j}^{k} + \Delta c_{j}^{k}, \qquad (13)$$

$$v_{j}^{k+1} = v_{j}^{k} + \Delta v_{j}^{k}. \qquad (14)$$

5. NEURAL NETWORK TRAINING ALGORITHM

The NN training algorithm has the following steps:

Step 1. Normalization of the input data. In the Elman's neural network, we select 9 parameters as input values and determine the learning speed η, which is between 0 and 1.

Step 2. Weights w_{ij} , c_j and v_j follows the distribution (-1, 1).

Step 3. We introduce a stochastic time-efficient function $\phi(t)$ into an error function E and choose a drift function $\mu(t)$ and a function that characterizes the tendency of changing the $\partial(t)$ network state. Elman network for predicting the time that overloads routes is shown in figure 2.

Step 4. While predicting the overload time of network overload, there may be a problem of so-called "dead neurons". One limitation of any competing layer is that some neurons may not be involved. That is, neurons with initial weight vectors are far removed from the entrance vectors and will never win competition, regardless of the term of training. As a result, such vectors are not used in training and the corresponding neurons never win (dead). Therefore, in order to provide the opportunity to defeat other neurons, the training algorithm provides for the possibility of losing a "neuron-winner" of its activity. For this purpose, the activity of the neurons is calculated on the basis of the calculation of the potential of each neuron in the process of forecasting the overload of data transmission paths and training of the neuron. First of all, the neurons of the layer are given a $p_i(0) = \frac{1}{c}$ potential, where c is the number of neurons (clusters).

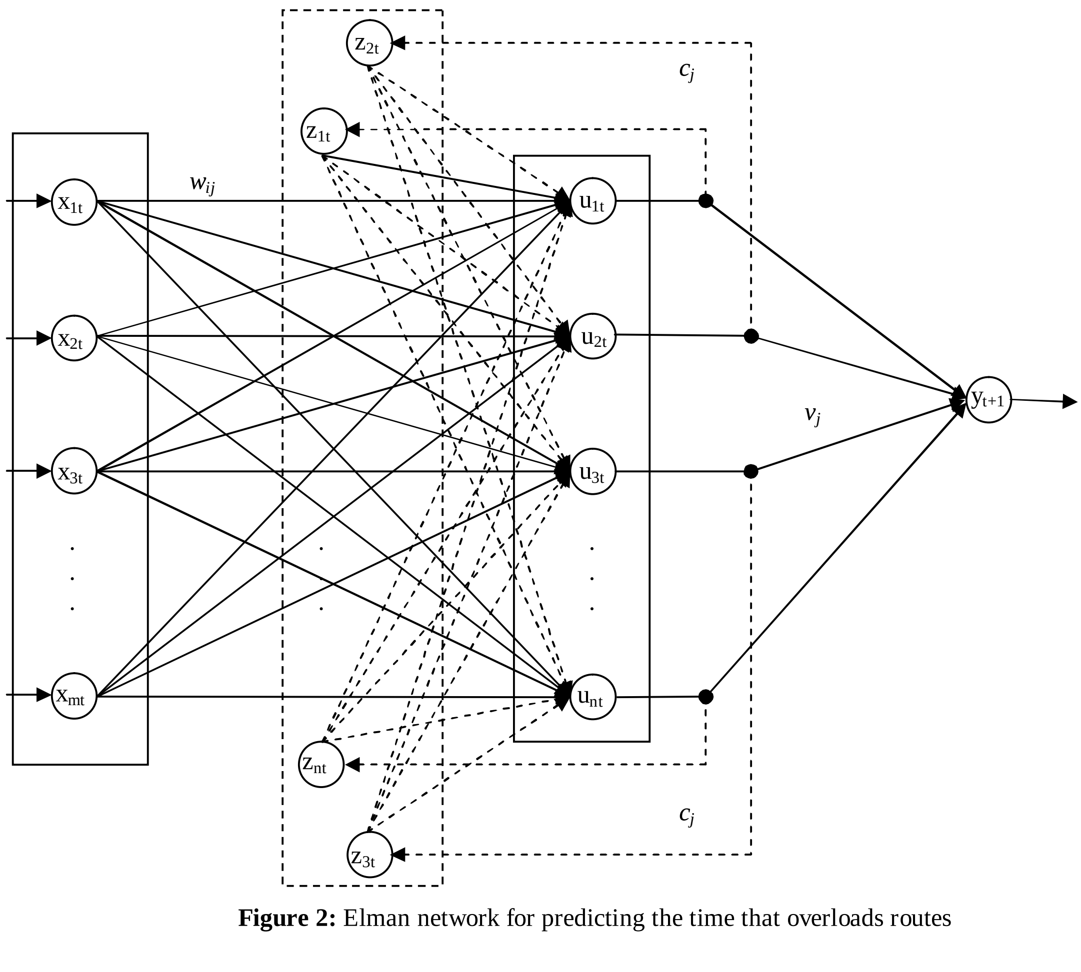

Figure 2: Elman network for predicting the time that overloads routes

If the value of the potential p_i falls below the $p_{\min}$ level of the neuron is excluded from consideration.

If $p_{\min} = 0$ neuron is not excluded from consideration.

If the $p_{\min} = 1$ neurons win one by one, as in each search cycle only one of them is ready to be considered.

In the k course of the training cycle, the potential is calculated according to the rule:

$$p_i(k) = \begin{cases} p_i(k-1) + \dfrac{1}{c}, & i \neq j, \\ p_i(k-1) - p_{\min}, & i = j, \end{cases} \tag{15}$$

where j is the number of "neuron-winner".

After providing equal opportunities for the victory of the neurons and calculating the error, the neural element of the winner with the k number will be determined:

$$d_k = \min_j d_j . \tag{16}$$

Step 5. Setting the minimum error ξ. If the value E is less than the specified minimum error, we pass on step 6, if we go further to step 7:

$$E = (1/N)\sum_{n=1}^{N} E(t_n) . \tag{17}$$

Step 6. Change the connecting weights: we calculate the gradient of connecting $w_{ij}, \Delta w_{ij}, v_j, \Delta v_j^k, c_j, \Delta c_j^k$ weights. Then we change the weights from the level to the previous $w_{ij}^{k+1}, v_j^{k+1}, \Delta c_j^{k+1}$ layer.

Step 7. Count the prediction value:

$$y_{t+1} = f_T\left(\sum_{j=1}^{m} v_j f_H\left(\sum_{i=1}^{n} w_{ij} x_{it} + \sum_{j=1}^{m} c_j z_{jt}\right)\right). \tag{18}$$

6. EVALUATE THE EFFECTIVENESS

The next step in developing the method is to evaluate the effectiveness of its operation. The conducted analysis showed that in order to determine the effectiveness of the methods of prediction of time series it is expedient to use the method of least squares [1, 2]. The essence of which is to find such linear and direct correlation coefficients, in which the sum of the squares of the deviations of the experimental data from the found will be the smallest. The method allows to calculate the linear value of the time series described by the equation of the $y_t = ax_t + b$ line, where a, b an unknown parameters of the time series model is, y_t, x_t is the value of the time series. Parameters are calculated by the formula:

$$a = \frac{n\sum_{i=1}^{n} x_t y_t - \sum_{i=1}^{n} x_t \sum_{i=1}^{n} y_t}{n\sum_{i=1}^{n} x_t^2 - \left(\sum_{i=1}^{n} x_t\right)}, \quad b = \frac{\sum_{i=1}^{n} y_t - a\sum_{i=1}^{n} x_t}{n}, \tag{19}$$

where n is the data amount.

In order to evaluate the effectiveness of the proposed forecasting method, methods for assessing the quality of the forecast should be used. The average forecast error value for the real data:

$$APE_k = \left|\frac{y_k - \tilde{y}_k}{y_k}\right| \cdot 100\% , \tag{20}$$

for the forecast period:

$$MAPE = \frac{1}{n}\sum_{k=1}^{n}\left|\frac{y_k - \tilde{y}_k}{y_k}\right| \cdot 100\% , \tag{21}$$

where n is the length of the period, y_k is the real data, $\tilde{y}_k$ is the forecast data.

7. CONCLUSION

The article presents the method of forecasting the time of overloading of data transmission routes in the MR, which is constructed using the neural network of Elman on the basis of the calculation of the potential of neurons in the network. Unlike existing forecasting methods that do not take into account the need for decentralization of management; decision making in conditions of uncertainty; reduction of the time of making a managerial decision, in the developed method, the specified features of the method's operation in the MP were taken into account.

This method allows to predict the time of overloading of data transfer routes in the MR due to the reduction of the computational complexity of the neural network and the application of the algorithm for training the neural network.

7. FUTURE SCOPE

In the course of further research, a methodology will be developed for applying methods for predicting time overload of data transmission routes and supporting data transfer routes based on knowledge base.

REFERENCES

1. Zhuk, P., Romanyuk, V., Sova O. and Bunin, S. “**Intellectual Mobile Ad Hoc Networks**”, In Proc. of International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET, Lviv, p. 238, 2012.

2. A. S. Mohammed "**Modification of load balancing method in networks with wimax technology**", Qalaai Zanist Journal, Vol. 3, No. 2, pp. 791–802.
3. Saravana Balaji B., Amin Salih Mohammed and Chiai Al-Atroshi, "**Adaptability of SOA in IoT Services – An Empirical Survey**", International Journal of Computer Applications, vol. 182(31), pp. 25-28, 2018, DOI: http://doi.org/10.5120/ijca2018918249 https://doi.org/10.5120/ijca2018918249
4. Kuchuk, G., Nechausov, S. and Kharchenko, V. "**Two-stage optimization of resource allocation for hybrid cloud data store**", International Conference on Information and Digital Technologies, IDT 2015, pp. 266-271, 2015, DOI: 10.1109/DT.2015.7222982 https://doi.org/10.1109/DT.2015.7222982
5. Mozhaev, O., Kuchuk, H., Kuchuk, N., Mykhailo, M. and Lohvynenko, M. "**Multiservice network security metric**", 2nd Int. Conf. on Advanced Information and Communication Technologies, AICT 2017 – Proc., pp. 133–136, 2017, DOI: 10.1109/AIACT.2017.8020083. https://doi.org/10.1109/AIACT.2017.8020083
6. Kara, Y., Boyacioglu, M. Acar and Baykan, O. K. "**Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange**", Expert Systems with Applications, vol. 38, no. 5, pp. 5311–5319, 2011. https://doi.org/10.1016/j.eswa.2010.10.027
7. Divya, P., and V. Manikandan. "**International Journal of Innovative Research in Engineering Science and Technology http://journal. selvamtech. com AN ENERGY SAVING TECHNIQUE FOR WIRELESS SENSOR NETWORKS**."
8. E. F. de Gorostiza, J. Berzosa, J. Mabe and R. Cortiñas "**A Method for Dynamically Selecting the Best Frequency Hopping Technique in Industrial Wireless Sensor Network Applications**", Sensors (Basel), 2018, DOI: 10.3390/s18020657. https://doi.org/10.3390/s18020657
9. Chaudhuri, T. D. and Ghosh, I. "**Artificial Neural Network and Time Series Modeling Based Approach to Forecasting the Exchange Rate in a Multivariate Framework**", Journal of Insurance and Financial Management, Vol. 1, No. 5, pp. 92–123, 2016.
10. Zhou, X., Zhuge, Q., Qiu, M., Xiang, M., Zhang, F., Wu, B., Qiu, K. and Plant, D. V. "**Bandwidth variable transceivers with artificial neural network-aided provisioning and capacity improvement capabilities in meshed optical networks with cascaded ROADM filtering**", Optics Communications, Vol. 409, pp. 23–33, 2018, DOI:10.1016/j.optcom.2017.09.021 https://doi.org/10.1016/j.optcom.2017.09.021
11. Neumann, D., Wiese, T. and Utschick, W. "**Learning the MMSE Channel Estimator**", IEEE Trans. on Signal Processing, 2018, DOI: 10.1109/TSP.2018.2799164.
12. AS Mohammad, My Potrus "**a Method For Compensation of Tcp Throughput Degrading During Movement of Mobile Node**" 2016.
13. V. Porkodi, Dr.D. Yuvaraj, Dr. Amin Salih Mohammed, V. Manikandan and Dr.M. Sivaram. "**Prolong the Network Lifespan of WirelessSensor Network**" (2018): 2034-2038.
14. Saravana Balaji, B., R. S. Raj Kumar, and N. K. Karthikeyan. "**Fuzzy Predicate Petri Net and T-invariant Analysis based Cloud Service Recommendation using Cloud Service Ontology.**" Publication of Book Journal, Mobile Computing A book of Readings (2015): 200-212.
15. GM Gouse, CM Haji, **Improved Reconfigurable based Lightweight Crypto Algorithms for IoT based Applications**, 186-193,
16. Ashish Sharrna "**A Smarter Way to Improvise Learning Outcome based on Computer Application**", 2019.
17. Rejab Hajlaoui, Tarek Moulahi1, Hervé Guyennet, "**Vehicular ad hoc networks: From simulations to real-life scenarios**", International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No,1.1, 2019, pp. 169-174 https://doi.org/10.30534/ijatcse/2019/3281.12019