INTERNATIONAL JOURNAL OF ELECTRONICS, MECHANICAL AND MECHATRONICS ENGINEERING Vol.6 Num.3 - 2016 (1241-1250)

Doi: 10.17932/IAU.IJEMME.m.21460604.2016.6/3.1241-1250

A Prediction Model For Performance Analysis in Wireless Mesh Networks

Safak Durukan Odabasi[1]
Ergun Gumus[2]

Abstract
Analysis of computer networks is an important study field that must be handled carefully in order to make communication systems work properly. Efficient evaluation and remodelling of system according to factors affecting the performance is required. For this aim, many techniques have been proposed, so far. However, machine learning methods are getting more preferable than others with their cost-effective and faster solutions. In this study, generalized regression neural networks (GRNNs) approach was employed in order to predict the output, packets dropped of a sample DMesh network simulation. The simulation is driven by parameters such as number of nodes, number of gateways, number of channels used, and traffic density. It was observed that parameters: traffic density and number of channels used, have a direct impact on error rate of the regression model. The high variance explained values show that GRNN approach can represent real characteristics of DMesh architecture.

Keywords: *Mesh networks, source management, prediction model, generalized regression neural networks, performance analysis.*

1. INTRODUCTION

The rapid development of network technologies necessitates the performance of systems be high and requirements of the users be provided appropriately. According to traffic pattern, a continuous data transmission or data integrity introduces the expectation of quality of service (QoS). In addition, limited sources like frequency require the source management be planned efficiently. Thus, analysis of system performance and designing of new mechanisms that fulfil system requirements effectively are necessities.

Performance analysis methods are generally simulations, testbeds and predictions for wireless networks. However, when

simulations or testbeds are used for analysis, a prior knowledge of all information about the system like the important parameters for network performance: bandwidth, error rate, jitter, throughput or latency, should be present [1]. At this point, usage of machine

[1] *Department of Computer Engineering, Istanbul University, 34320, Avcilar, Istanbul, Turkey, sdurukan@istanbul.edu.tr*
[2] *Department of Computer Engineering, Istanbul University, 34320, Avcilar, Istanbul, Turkey, egumus@istanbul.edu.tr*

learning techniques to predict system performance becomes a cost-effective approach. Prediction algorithms need only small amount of information that makes them result faster than other methods. They can be efficiently used to coordinate and optimize network parameters according to changes in traffic where manual adaptation is so difficult due to variety of different type networks [2].
Fast development of data centres increases density of non-real-time data traffic like data backup information transmission and correspondingly causes fluctuations on real-time user's traffic. To prevent this, traffic prediction algorithms can make the non-real-time traffic be transmitted at the time when real-time traffic density is low. In the light of all these advantages, machine learning techniques become more preferred than the other ones for next generation network systems.

In this study, a Generalized Regression Neural Network (GRNN) based prediction model is presented for analysis of DMesh network simulation which is one of the rising next generation network architectures with its numerous advantages [3]. The prediction model estimates *packets dropped* rate in the network simulation using various inputs like *number of gateways*, *number of nodes*, *number of channels used*, and *traffic density*. Predictions of the model are compared to actual outputs of the simulation. Results prove that by using sufficient number of observations, GRNN based prediction model can represent real characteristics of DMesh architecture only with tolerable amount of error. Throughout the study, topics like the relation between traffic density and prediction error, and determining the required number of observations for prediction are also issued.

Rest of the paper is organized as follows: Section 2 presents some of previous studies about prediction techniques used for network performance analysis. Section 3 gives brief information about DMesh architecture, its simulation and GRNN approach. Experimental results are given in Section 4. Finally, the study is concluded in Section 5.

2. STUDIES ON TRAFFIC ANALYSIS

Machine learning techniques became popular for performance evaluation of computer networks because of their significant advantages. There are many studies that benefit from machine learning techniques.

Machine learning techniques are invoked frequently and new methods are proposed since maintenance and operation of network is crucial for Software Defined Networks (SDNs) which have complex network traffic. EMD-based multi-model prediction (EMD-MMP) [2] algorithm that is proposed for short-term traffic forecasting combines traditional prediction methods with the EMD to improve the network prediction accuracy by referring characteristics of EMD for simplifying complicated data.

One of the most important challenges for network analysis is link prediction. It is used to detect illegal and hidden organizations at social security networks while human behaviour is analyzed at social networks. In [4], link prediction problem at probabilistic temporal uncertain networks is handled. Studies using machine learning for link prediction are analyzed and a new method based on random walk algorithm is proposed. This new method combines temporal and global topological information with higher quality than existing studies.

Environmental monitoring is a popular application example of wireless networks. Constraints like battery life cause scaling problems while environmental monitoring with the help of Wireless Sensor Networks (WSN). Three processing steps are followed during environmental monitoring on a WSN: prediction, compression and recovery. A new framework, proposed in [5], compounds these steps. Least mean square (LMS) is used for data prediction at both node and cluster head, then central Principal Component Analysis (PCA) is used for data compression. Finally, base station recovers original data with error tolerance. Combination of these three steps makes this framework cost-effective.

There are various factors affecting performance of a network such as network size, mobility of network, and so on. Therefore, design of routing protocols must be handled carefully. During this design procedure, network behaviour is needed to be analyzed, efficiently. It is possible to associate protocol performance with metrics by using regression models. In [6], an adaptive control method that uses Protocol Regression Model (PRM) to select most suitable routing algorithm for the case network is proposed. By this way, instead of designing a new protocol, it is proved that existing protocols can be used effectively where a unique routing protocol fails for all possible environmental conditions and requirements.

WMNs have infrequency in terms of traffic change since they have a large number of end users. This characteristic of WMN makes traffic classification become complicated. In [7], an online traffic classification tool is developed. Semi-supervised architecture of the tool is its strong suit and makes it possible to achieve high performance with less data samples.

3.MATERIALS AND METHODS

This section presents brief information about the Mesh network simulation and regression methodology subject to this study.

3.1. DMESH (DIRECTIONAL MESH ARCHITECTURE)

Wireless Mesh Networks (WMNs), a kind of multi-hop ad hoc networks, are in place among next generation networks with their significant benefits such as easy maintenance, high security, self-configuration, low cost and robustness [8, 9]. Features like broadband access and rapid fixing of connection failures make them usable as emergency or disaster communication systems [10].

A typical WMN consists of three layers: gateways, mesh routers and clients.

Gateways work as bridges to connect WMN to other networks. Mesh routers that are responsible for receiving/transmitting data packets from/to other networks, have special abilities in addition to basic ones. Mesh clients are combinations of fixed and wireless mesh devices that use WMN services. Mesh clients may compose a client mesh network among themselves or with mesh routers.

In multi-channel multi-radio (MC-MR) WMNs systems, each mesh router can be equipped with multiple antennas to increase network overall capacity. Thus, a router that equipped with various radio interfaces can communicate with multiple routers simultaneously. A well-planned channel assignment (CA) algorithm is responsible for adjustment of each antenna to different channels. By this way, it provides minimization of interference between the channels and ensures setting up proper data paths between the nodes.

DMesh is the first architecture that uses directional antennas with an omnidirectional antenna to the best of our knowledge [3]. Usage of inexpensive and easy-setup feature directional antennas brings on DMesh the best performance among similar architectures. DMesh ensures effective frequency usage by its conservative channel assignment scheme while it increases the inference level of the network.

In this section, dynamic and distributed CA scheme that is used by DMesh architecture and called as C-DCA is proposed. C-DCA is a dynamic and distributed CA method that aims to increase throughput of MC-MR WMNs. DMesh combines spatial separation in directional antennas with frequency separation in orthogonal channels. In this way, more transmissions with less interference are achieved. Besides, DMesh benefits from the advantages of practical directional antennas like inexpensiveness and wide beamforming.

There are several studies to improve throughput of MC-MR WMNs in the literature [11, 12]. However, in these studies, just omnidirectional antennas are used on the routers that increases interference level of the network. That is, increasing throughput while decreasing interference goal of CA schemes fails. On the other hand, DMesh overcomes this dilemma with its distributed and dynamic CA scheme (C-DCA). DMesh follows three steps on CA procedure: composing a physical tree of which root(s) is/are gateway node(s), routing packets through the network and performing CA scheme.

The routing process is called as Directional Optimized Link State Routing (DOLSR), which is an extended version of Optimized Link State Routing (OLSR) [13] obtaining multi-hop routes in single-radio single-channel omnidirectional mesh networks [3].

3.2. GENERALIZED REGRESSION NEURAL NETWORKS (GRNNS)

Artificial Neural Networks (ANNs) are layered structures consisting of interconnected nodes called “neurons”, inspired by biological neurons. Connection between each node pair is rated with tunable weights which are adjusted by a series of input patterns and their corresponding outputs. This adjustment process is known as "learning" or "memorizing", and achieved by using various learning rules [14]. With their customizable structure, ANNs have the valuable property of generalization for revealing complex relations between inputs (*X*) and targeted outputs (*Y*).

ANNs can be employed for a wide range of learning tasks. In this context, Generalized Regression Neural Networks (GRNNs) [15, 16] are their specialized versions for regression. GRNNs are a type of radial basis networks used for function approximation. They are usually composed of two layers: input layer (radial basis layer) and output layer (linear layer). Input layer produces the net input $\Sigma_i = \| X_p - W_i \| b(i)$ of *i*th neuron, where $b(i)$ is the bias term, and $\| X_p - W_i \|$ is the Euclidean distance between the input pattern X_p and weight vector W_i of *i*th neuron. The net input is fed to radial basis function $f(x) = e^{-x^2}$, normalizing each output of first layer to 0-1 range. The output is obtained by application of linear transfer function $g(x) = x$ on net input of the output layer. A typical GRNN is shown in Fig. 1. Here, X_p and Y_p correspond to *p*th input/output patterns.

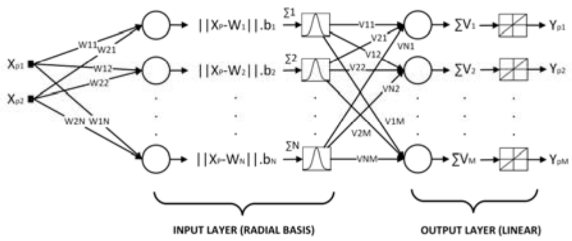

Figure 1. General form of generalized regression neural network.

3.3. DMESH SIMULATION

DMesh simulation was prepared using Matlab [17]. Specifications of the simulation are presented in Table 1.

The network simulation is driven by four parameters: *number of gateways*, *number of nodes*, *number of channels used*, and *traffic density* affecting the outcome *packets dropped rate* (%).

Number of gateways, *number of nodes*, and *number of channels used* are predetermined at the beginning of the simulation. Besides, the location information (coordinates) and deviation angles of each node are set randomly. After forming the network logically, routing trees are set up and CA is handled.

Table 1. Specifications of DMesh simulation.

Simulation area	500m × 500m
Number of gateways	Varies between 1-10 (incremented by 1)
Number of nodes	Varies between 10-200

	(incremented by 10)
Number of usable channels	3, 6 or 12
Deviation angle of a directional antenna	Chosen randomly between [$0, \frac{\pi}{2}$]
Traffic Density	Varies between 10%-100% (incremented by 10%)
Traffic model	Poisson
Packet Size	1500 bytes
Bit rate	54 Mbps
Simulation time	100 sec

A typical network architecture formed using the simulation is illustrated in Fig. 2.

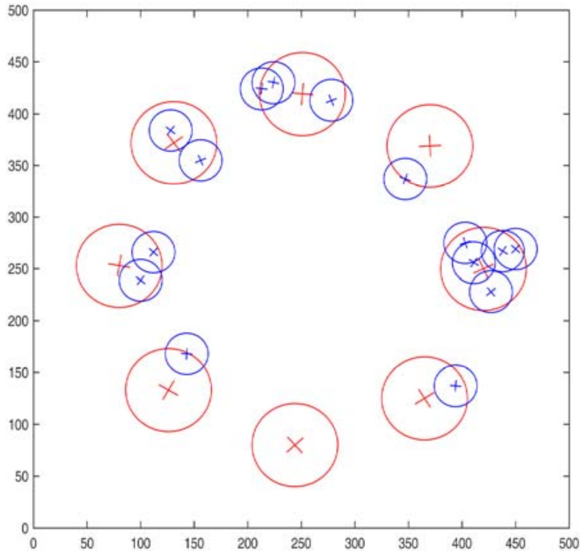

Figure 2. A 12-channel Mesh topology with 8 gateways
(red circles) and 15 nodes (blue circles).

4. RESULTS

In order to evaluate the performance of GRNN predicting the drop rate, three distinct regression networks were created with *number of gateways* and *number of nodes* as the inputs and *packets dropped* rate as the output. Each one of the three networks corresponded to a simulation with specific number of channels (3, 6, or 12) used. The data set contained 6000 observations which were split into ten distinct groups according to *traffic density* parameter and half of the samples from each group were fed into corresponding network for training. After training process, remaining half of the samples were fed into the network and predictions were obtained. In order to eliminate the bias effect, this train-test process was repeated 100 times for each group using random permutations of samples for training/testing. Mean values of outputs (*packets dropped rate*) for each test sample were calculated and stored as predictions of corresponding GRNN. Fig. 3 depicts observations and predictions of the three GRNNs with the *traffic density* parameter of 50%.

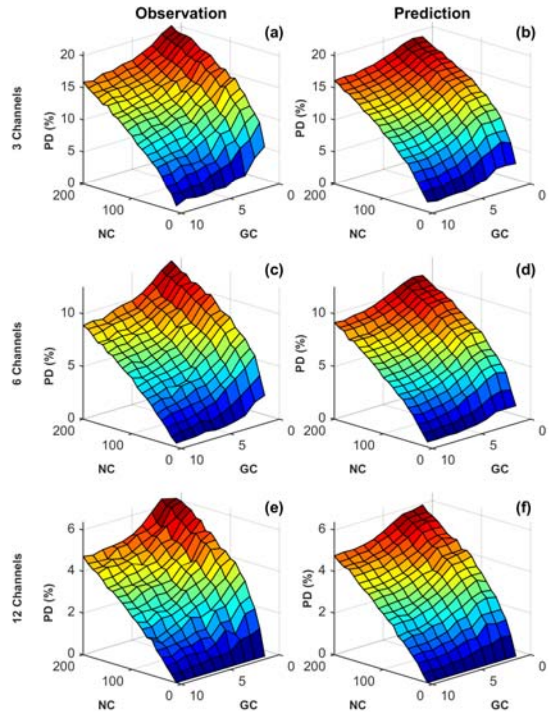

Figure 3. Observations and corresponding predictions for various channels (NC: Node count, GC: Gateway count, PD: Packets dropped).

As can be seen from Fig. 3(a)(c)(e), *packets dropped* rate falls as the *number of channels used* rises. This is directly a result of reduced number of collisions. Besides, it can also be stated that *number of nodes* (NC) and *number of gateways* (GC) parameters have opposite effects on *packets dropped*. Coming to our main concern, Fig. 3(b)(d)(f) verify that predictions of GRNNs present same characteristics as observations which can be seen from explained variance values given in Table 2. High values are indicators of goodness of fit. The characteristics seem not to depend on *traffic density*.

Table 2. Explained variance of trained GRNNs.

Traffic Density	**Variance Explained**		
	3 Channels	**6 Channels**	**12 Channels**
10%	0.9611	0.9628	0.9697
20%	0.9604	0.9641	0.9691
30%	0.9617	0.9641	0.9692
40%	0.9612	0.9632	0.9692
50%	0.9614	0.9623	0.9685
60%	0.9613	0.9626	0.9687
70%	0.9616	0.9627	0.9680
80%	0.9605	0.9643	0.9682
90%	0.9607	0.9627	0.9693
100%	0.9608	0.9636	0.9688

Although having same characteristics, predictions are not identical to observations which are the main concept of “generalization”. Fig. 4 presents the variation in prediction error (Root Mean Squared Error) according to the change in *traffic density*. The *packets dropped* rate rises as the *traffic density* rises, yielding to more diverse observations and an increment in RMSE. On the other hand, the reduction in *number of channels used* increases *packets dropped* rate which explains higher RMSE values obtained from GRNNs of smaller *number of channels.*

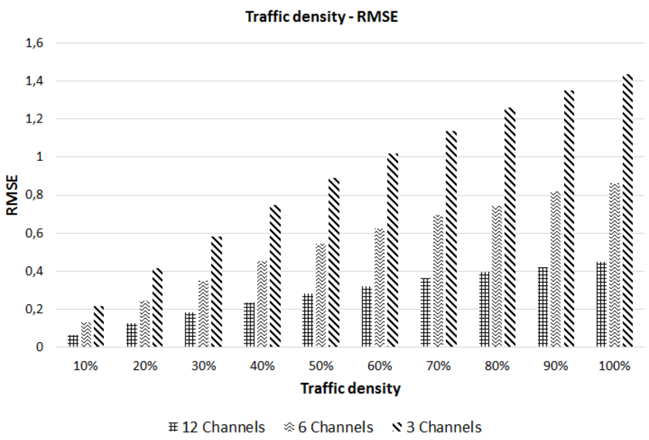

Figure 4. Variation in RMSE according to traffic density.

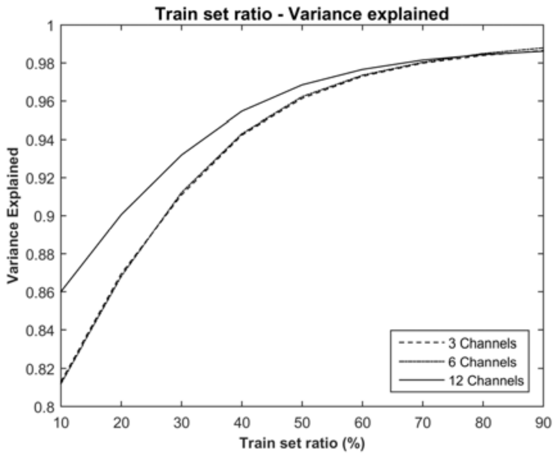

Figure 5. Train set ratio - Variance explained.

As stated before, presented results were obtained by using 50% of samples for training and the remaining 50% for testing, in several trials. This choice of split provided a high explained variance value of 0.96 (see Table 2) for all of the three GRNNs meaning that predicting the *packets dropped* rates of *N* cases could be achieved by using *N* observations. As might be expected, changing the ratio of train set size can affect the generalization performance. Fig. 5 presents the change in variance explained according to increasing ratio of training set size. It can be clearly seen that the GRNNs that were trained by samples of 3 and 6 channels exhibit same manner and their explained variance values overlap. However, the GRNN for 12 channels presented a better generalization performance. All of the three networks meet at explained variance value of 0.98, when using 70% of samples for training, which seems to be a peak. Some might also choose this as preferred train split ratio.

5. CONCLUSIONS

Determination of the factors that affect the performance of a network system is substantial in the way of facilitating to make proper regulations. A method for evaluation of network performance needs to be successful in addition to providing an efficient approach in terms of speed and cost. In this context, using machine learning methods offers faster and cost-effective results using only small amount of data in order to predict system behaviour. This provides these methods be mentioned along with other methods for network performance analysis.

In this study, generalized regression neural networks (GRNNs) approach was employed in order to predict the output, *packets dropped,* of test cases. Although, four inputs determine the output, only *number of gateways* and *number of nodes* were used as inputs of separate networks in order to evaluate effect of *number of channels used* and *traffic density* on regression. The *traffic density* parameter did not seem to influence the explained variance, however, it had a direct impose on RMSE due to its effect on output. On the other hand, *number of channels used* parameter also had a direct impose on RMSE, but due to its inverse proportion to the output, this impose was also in opposite way.

In addition to these, effect of train/test split was also examined. Results proved that, by choosing an ideal proportion for training, GRNNs can provide high explained variance and low RMSE values which are indicators of goodness of fit. This makes them good candidates for estimating output parameters of DMesh architecture.

Acknowledgements Authors thank Assoc. Prof. Dr. Olcay Kursun from the same department for helpful discussions.

REFERENCES

[1] Rappaport T.S., *Wireless communications: principles and practice (2nd edition)*. Upper Saddle River, NJ: Prentice Hall PTR. ISBN 0-13-042232-0, 2002.

[2] Dai L., Yang W., Gao S., Xia Y., Zhu M., and Ji Z., "EMD-based multi-model prediction for network traffic in software-defined networks," *in IEEE 11th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, Philadelphia, PA, pp. 539-544, 2014.

[3] Das S.M., Pucha H., Koutsonikolas D., Hu C., and Peroulis D., "Dmesh: Incorporating practical directional, antennas in multi-channel wireless mesh networks," *in IEEE J. Sel. Areas Commun.*, vol. 24, no. 11, pp. 2028-2039, 2006.

[4] Ahmed N.M., and Chen L., "An efficient algorithm for link prediction in temporal uncertain social networks," *Information Sciences,* vol. 331, pp. 120-136, 2016.

[5] Wu M., Tan L., and Xiong N., "Data prediction, compression, and recovery in clustered wireless sensor networks for environmental monitoring applications," *Information Sciences,* vol. 329, pp. 800-818, 2016.

[6] Priya S.B.M., "Adaptive control of routing protocol in mobile adhoc network using regression model," *in International Conference on Emerging Trends in Science, Engineering and Technology (INCOSET)*, Tiruchirappalli, Tamilnadu, India, vol. 13-14, pp. 509-514, 2012.

[7] Gu C., Zhang S., Xue X., and Huang H., "Online wireless mesh network traffic classification using machine learning," *Journal of Computational Information Systems,* vol. 7, no. 5, pp. 1524-1532, 2011.

[8] Alzubir A., Bakar K.A., Yousif A., and Abuobieda A., "State of the art, channel assignment multi-radio multi-channel in wireless mesh network," *International Journal of Computer Applications,* vol. 37, no. 4, pp. 14-20, 2012.

[9] Riggio R., Pellegrini F.D., Miorandi D., and Chlamtac I., "A knowledge plane for wireless mesh networks," *Ad Hoc & Sensor Wireless Networks,* vol. 5, pp. 293-311, 2007.

[10] Yarali A., Ahsant B.., and Rahman S., "Wireless mesh networking: a key solution for emergency&rural applications," *in IEEE Second International Conference on Advances in Mesh Networks*, pp. 143-149, 2009.

[11] Draves R., Padhye J., and Zill B., "Comparison of routing metrics for static multi-hop wireless networks," *in Conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM),* pp. 133-144, 2004.

[12] Raniwala A., and Chiueh T., "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," *in 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, New York, pp. 2223-2234, 2005.

[13] https://tools.ietf.org/html/rfc3626

[14] Zurada J.M., *Introduction to artificial neural systems.* West Publishing Co., St. Paul, MN, USA, 1992.

[15] Specht D.F., "A general regression neural network," *IEEE Transactions on Neural Networks*, vol. 2, no. 6, pp. 568-576, 1991.

[16] Wasserman P.D., *Advanced methods in neural computing.* New York, Van Nostrand Reinhold, pp. 155-61, 1993.

[17] MATLAB Release 2015a, The MathWorks, Inc., Natick, Massachusetts, United States.