

Conference Paper

Decision Support System for Alarm Correlation in GSM Networks Based on Artificial Neural Networks

Ashraf Kamal Arhouma¹ and Saleh M. Amaitik²

¹ *Mobile Communication Group Department, Alcatel-Lucent, Benghazi, Libya*

² *Industrial and Manufacturing Systems Engineering Department, University of Benghazi, Benghazi, Libya*

Correspondence should be addressed to Ashraf Kamal Arhouma; a.arhouma@gmail.com

Received 27 February 2013; Accepted 9 May 2013

Academic Editors: M. Elmusrati, A. Gaouda, and H. Koivo

This Conference Paper is based on a presentation given by Ashraf Kamal Arhouma at “International Conference on Electrical and Computer Engineering” held from 26 March 2013 to 28 March 2013 in Benghazi, Libya.

Copyright © 2013 A. K. Arhouma and S. M. Amaitik. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As mobile networks grow in size and complexity, huge streams of alarms are flooding the operation and maintenance center (OMC). Thus, the operator needs a decision support system that converts these massive alarms to manageable magnitudes. Alarm correlation is very important in improving the service and the efficiency of the maintenance team in mobile networks and in modern telecommunications networks. As any fault in the mobile network results in a number of alarms, correlating these different alarms and identifying their source are a major problem in fault management. In this paper, an artificial neural network model is proposed to interpret the alarm stream, thereby simplifying the decision-making process and shortening the operator’s reaction time. MATLAB program is used as programming tool to develop, implement, and compare between different types of designed artificial neural network models. To assist the operators to take fast decision and detect the root cause of the alarms, the alarms and the result of the artificial neural networks model are visualized in real time on the Google Earth application.

1. Introduction

A medium-sized telecom network operations centre receives several hundred thousand alarms per day. This volume of alarms creates severe challenges for the operations and management staff [1]. Alarms are unsolicited messages received from network elements or network management systems carrying information about malfunction location, type, severity and other relevant information of the mobile network sites. Usually, alarm information consists of categorical and numerical attributes and is difficult to be processed by most of clustering algorithms [2].

The alarm correlation is an essential function of network management systems to provide detection, isolation, and correlation of unusual operational behavior of telecommunication network. However, existing alarm correlation approaches still rely on the manual processing and depend on the knowledge of the network operators [3]. Alarm correlation analysis system is useful method and tool for

analyzing alarms and finding the root cause of faults in telecommunication networks [4].

The main objective of this research work is to develop a model which provides the engineers, managers and OMC operators with a decision support system to analyze the alarms and find the root cause of the alarms burst in the GSM network, so that they will be able to make correct decisions, save time during the diagnoses, and provide correct guidance to the maintenance team.

This paper consists of nine sections. The next section shows the related work, and the Section 3 contains an overview of the mobile network architecture, and in Section 4 the problem is described. In Section 5 the ANN model is presented and Section 6 introduces the case study. Section 7 discusses the results, and Section 8 shows how to display the inputs and outputs of the ANN model in Google Earth application. Section 9 contains the conclusions of the presented work.

2. Related Work

Wietgreffe et al. (1997) presented an Artificial neural network based alarm correlation system which uses a Cascade Correlation neural network to correlate alarms in a GSM network. The results of that approach show that the Cascade Correlation Alarm Correlator (CCAC) is well suited for alarm correlation tasks. The behavior in the case of noisy data (additional/missing alarms) is discussed and compared in detail to a codebook approach [5].

Wietgreffe (2002) presented in Network Operations and Management Symposium (NOMS) a paper that compares and assesses several alarm correlation methods for their suitability and performance in global systems for mobile communications (GSM). Out of the neural networks investigated, the cascade correlation learning algorithm performs best. This approach is compared with correlation techniques proposed in literature: rule-based diagnosis, model based diagnosis and alarm correlation using codebooks. It is shown that for alarm correlation in a GSM access network the proposed cascade correlation approach is superior to the other correlation techniques [6].

Nowadays, communication network turns to be more complex; once there occurs a failure, it will result in multi-alarm events which require relevant transactions. Jian and Ming (2008) showed that both theoretical analysis and computer simulations illustrate outstanding performance of the proposed models, which can be further optimized by experiments for specific environment [7].

Jukic and Kunštic (2010) describe in their paper an architecture proposal for Alarm Basic Correlations Discovery Environment and also discuss some implementation aspects [8].

According to Kim et al. (2011), the telecommunication network produces a number of alarms which are so-called the alarm floods, which is very difficult to detect the root cause problems. Therefore, alarm correlation algorithm was proposed which is able to isolate and correlate the root causes in a very short time. In addition, the proposed algorithm performs well in terms of efficiency of analyzing alarms and accuracy of identifying root cause [3].

Chao (2011) introduces a method of mining correlation rules of alarm messages by modified adaptive resonance theory network (MART) algorithm. The experiments illustrate good performance of the proposed algorithm with concept hierarchy tree to solve similar degrees of mixed data [2].

3. Architecture of Mobile Network

The global system of mobile communication (GSM) network is composed of several functional entities, whose functions and interfaces are specified [9]. Figure 1 shows the layout of a generic GSM network. The GSM network can be divided into three broad parts.

- (1) The mobile station is carried by the subscriber.
- (2) The access network controls the radio link with the Mobile Station.

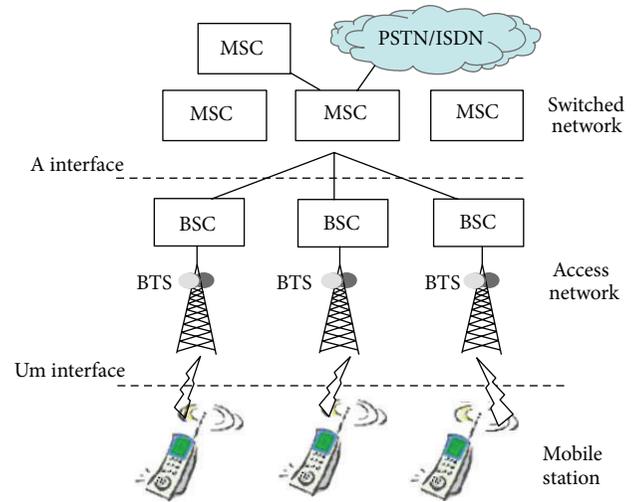


FIGURE 1: GSM architecture.

- (3) The Switched network, the main part of the Switching Subsystem is the Mobile Services Switching Center (MSC) which performs the switching of calls between the mobile users and between mobile and fixed network users.

All three parts of the GSM network are operated and managed through the Operation and Maintenance Centre (OMC).

The mobile station and the access network communicate across the air interface (Um interface), also known as radio link. The access network communicates with the Mobile Services Switching Center across the A interface.

The mobile station (MS) consists of the mobile equipment (the terminal) and a smart card called the Subscriber Identity Module (SIM). The SIM provides personal mobility, so that the user can have access to subscribed services irrespective of a specific terminal.

The access network is composed of two parts, the Base Transceiver Station (BTS) and the Base Station Controller (BSC). These communicate across the standardized Abis interface, allowing (as in the rest of the system) operation between components made by different suppliers.

The Base Transceiver Station houses the radio transceivers that define a cell and handles the radio-link protocols with the mobile station. In a large urban area, there will potentially be a large number of BTSs deployed; thus the requirements for a BTS are ruggedness, reliability, portability, and minimum cost.

The Base Station Controller manages the radio resources for one or more BTSs. It handles radio-channel setup, frequency hopping, and handovers. The BSC is the connection between the mobile station and the Mobile Service Switching Center (MSC).

The central component of the switched network is the Mobile Services Switching Center (MSC). The Home Location Register (HLR) and Visitor Location Register (VLR), together with the MSC, provide the call-routing and roaming capabilities of GSM.

4. Problem Description

In the access network of GSM systems, several BTS are connected via a multiplexing transmission system to BSC. These connections are very often realized with microwave (MW) line-of-sight radio transmission equipment. Heavy rain, dust or wind can temporarily disturb the connections between the antennas. The temporary loss of sight of a microwave disconnects all chained BTS from the BSC and results in an alarm burst [6].

Figure 2 shows the logical star topology connecting a BTS to the BSC is physically a tree network, where the traffic to the BTS is distributed over a chain of microwave systems and leased lines. For example, the logical connection between the BSC and BTS4 is provided via BTS1, BTS2, and BTS3. Mainly for cost reasons, there is only one path between a BTS and the controlling BSC.

When a link fails (e.g., between BTS1 and BTS2 in Figure 2), many alarms are generated and passed to the Operation and Maintenance Center (OMC). These alarm bursts have major impacts to the network management

- (1) Stressful conditions: they create stressful conditions for the network management staff that has to deal with all the alarms [10].
- (2) Reduce the efficiency of the maintenance team: it could be very difficult for the network operators to detect the root cause problems in a short period of time [3] especially for nonexpert engineers in the OMC and that will lead to wrong diagnoses which affect the efficiency of the maintenance team.
- (3) Waste of time: for expert engineers sometimes it takes a long time to detect the main causes of those alarms especially if there are more than one link failed which cause a time waste.

Observations in the network show that during those alarm bursts alarm messages or other management event reports disappeared somewhere in the transmission path from the issuing network element to the OMC. The loss of event reports is a serious problem for the network management. First, important events might be disappeared, and second, tools used for alarm correlation have to be designed in a manner that they can cope with the fact that not all expected alarms will arrive at the OMC [11].

5. Proposed Artificial Neural Network Model

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, have the ability to deal with problems that are generally characterized by nonlinearities. It can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques and can be used when there is no exact knowledge about the relationships between the various parameters of the problem. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze.

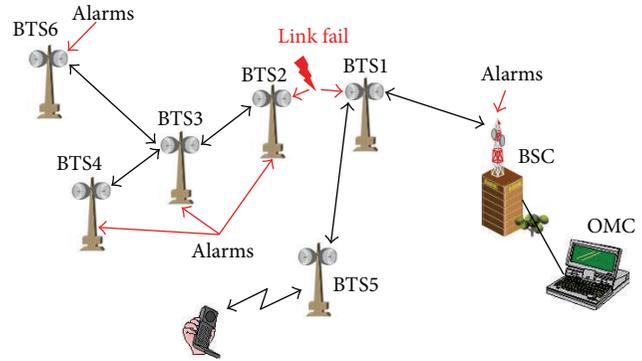


FIGURE 2: Alarms caused by a link fail.

So the neural network approach was chosen for this paper because of the following.

- (i) No expert knowledge is needed to train the neural network, neither for the initial configuration of the access network nor for its adaptation.
- (ii) If the configuration of the network is modified, ANN allows the system to adapt itself as a result of changing.
- (iii) They have the ability to deal with problems that are generally characterized by missing information, so it is useful in case we have some alarms disappears during the alarm burst.

The ability to learn from experience as well as the property of generalization makes ANN a very effective and powerful tool for alarm correlation in GSM networks.

The developed ANN model will be used as decision support system to diagnose the alarm correlation and find the root cause of the alarms burst in GSM network. The proposed model can deal with the fact of losses of some expected alarms. In order to achieve this goal, the following specific tasks have been accomplished.

- (1) Preparing the dataset to be used in the development of the proposed model. This process involves collecting the alarms generated on the OMC and the actual failed links, so it could be used as inputs and outputs to the proposed model. It is important that the data cover the range of inputs for which the network will be used.
- (2) Developing the supervised ANN model. BTS's alarm is represented by one neuron at the input layer. Each initial cause is represented by a neuron in the output layer. So each active alarm is represented as "+1" value at the corresponding input neuron and the state "no alarm" is noted as "0" value at the corresponding input neuron. The related faulty MW links are represented as "+1" at the corresponding output neuron and are represented as "0" value at the corresponding output neuron if the MW links are not faulty. Figure 3 shows the presentation of the BTS alarms and MW links fail in the ANN model.

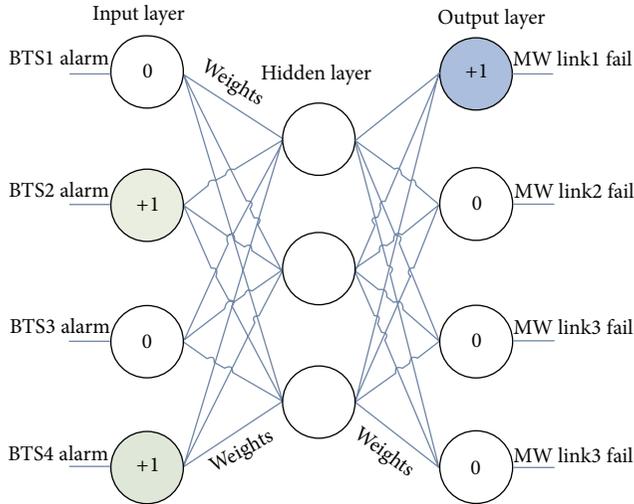


FIGURE 3: Presentation of ANN model with BTS Alarms and MW links fail.

The framework of the ANN application will pass through three phases:

- training phase, which includes exposing the designed ANN to a collection of representative examples (inputs and outputs) until the network learns enough from these examples and can produce the expected outputs. During training, the weights of the connections are trained and adapted (see Figure 3);
- validation phase, which includes exposing the designed ANN to a collection of new representative examples (inputs and outputs), and when the performance with the validation test stops improving, the algorithm halts (early stopping) and the network with the best performance on the validation set is then used for actual testing;
- testing phase, which includes exposing the trained network to a subset of inputs not previously seen by the network and comparing the predicted outputs with the actual outputs in order to check the validity of the trained network.

6. Case Study

MATLAB program and its Neural Network Toolbox are used for designing, implementing, visualizing, and simulating the neural network during the three phases of the development of the ANN model. A real GSM network, ALMADAR network in the east part of LIBYA, has been chosen to be a case study.

ALMADAR's network is divided into different subareas, each area consists of a group of BTS sites and MW links and each area can be represented by one neural network to facilitate its solution and allows the execution of several correlation processes in parallel, increasing the speed of the overall correlation process and reducing the complexity of the problem [12].



FIGURE 4: BTS sites and MW links.

TABLE 1: Allocation of training, testing, and validation sets.

Training set (70%)	Validation set (15%)	Testing set (15%)	Total
39	8	8	55

Figure 4 shows an example of those areas. This area which is chosen consists of 10 BTS sites and 9 MW links and the tenth one (not in Figure 4) is the main link to the BSC.

Each BTS from the ten BTSs will be presented by one neuron at the input layer and each MW link is represented by a neuron in the output layer.

A training sample of 55 patterns (each pattern formed by input and output vectors) has been prepared for training. It is important that the data cover the range of inputs for which the network will be used. The data is divided into three subsets. The first subset is the training set, which is used for computing the error and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process. The validation error normally decreases during the initial phase of training, as does the training set error. However, when the network begins to overfit the data, the error on the validation set typically begins to rise. The network weights and biases are saved at the minimum of the validation set error. The third subset is the test set, which is not used during training, but it is used to compare different models. The default ratios in the MATLAB program for training, testing, and validation data patterns are used. The numbers of data patterns allocated to the training, testing, and validation sets are shown in Table 1.

Several training experiments with various combinations of training parameters have been carried out to identify the optimal network structure and configuration which produce minimum errors during training phase. Due to the short time which did not allow carrying out large number of experiments covering various combinations of network configuration and training parameters, so the following combinations are used in the training.

- Two different types of neural networks were used: the feed-forward network and cascade-forward network.
- In the experiments, different numbers of hidden layers were used. In some experiments we used single hidden layers and in others we used two hidden layers.

TABLE 2: Training error limit, iterations limit, and validation stop values.

Type	Error limit	Iteration limit	Validation stop
Value	$1e - 10$	1000	6

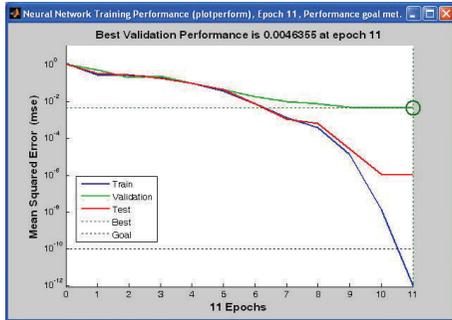


FIGURE 5: Performance progress for experiment 1.

- (iii) The numbers of hidden neurons in each hidden layer were 10 and 20 neurons.
- (iv) We used 3 different training algorithms from a list of the training algorithms that are available in the Neural Network Toolbox software in MATLAB program.

It has been considered that the network has learned the representative examples well enough when the error reaches a value lower than a predetermined limit, or the number of training cycles reaches a value equal to a predetermined limit, or the training continued until the validation error failed to decrease for six iterations (validation stop). Table 2 presents the training error limit, iterations limit, and validation stop values.

Twenty-four training experiments have been carried out as shown in Table 3. The table can be divided into two groups (Feed-Forward and Cascade-Forward) and each group is divided into two subgroups (one hidden layer and two hidden layers).

7. Results and Discussions

The two experiments that produced lowest mean square error (mse) value were selected and marked with asterisk which are experiment number 1 ($mse = 1.07337e-012$) and experiment number 19 ($mse = 1.65931e-012$) to verify its performance.

The graphs shown in Figures 5 and 6 present the performance progress which represent the training Mean Square Error (mse) on the y -axis against the number of epochs (training cycles) elapsed on the x -axis. Epochs represent a complete pass through the network of the entire set of training patterns.

The graphs generally illustrate downward movement of the error rate as learning progressed, indicating that the average error decreased between actual and predicted results.

Noting that experiments nos. 1 and 19 have achieved to its minimum performance value (stopped by performance goal

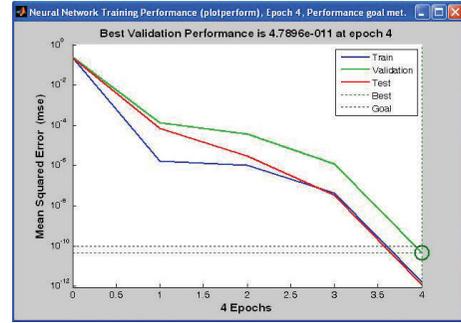


FIGURE 6: Performance progress for experiment 19.

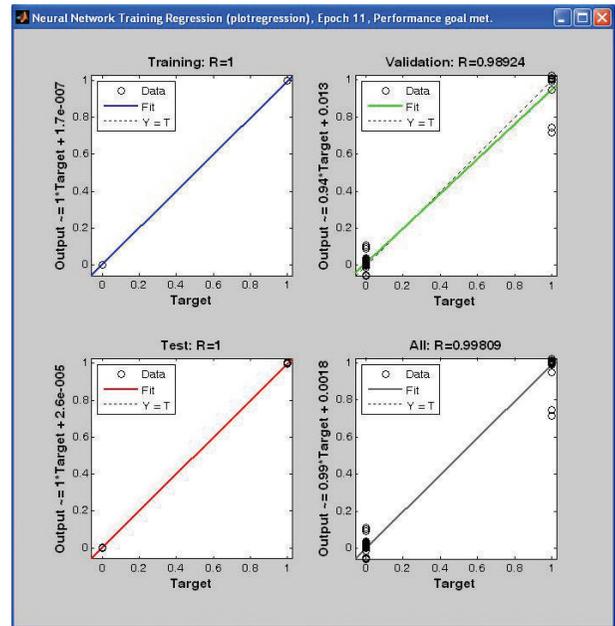


FIGURE 7: Regression plot for experiment no. 1.

met) but experiment no. 1 was the least but not with a big difference.

As can be seen, the learning time obtained in experiments 1 and 19 was so fast (finished in 11 and 4, resp.). It does not appear that any significant overfitting has occurred in the experiments and it can be noted that the test error and the validation error have the similar characteristics.

The next step in validating the network is to check the regression plots, which show the relationship between the outputs of the network and the targets and the results are shown in Figures 7 and 8.

The three axes represent the training, validation, and testing data. The dashed line in each axis represents the perfect result (outputs = targets). The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. As shown in Figures 7 and 8 the training and test data in the two experiments indicates a

TABLE 3: Training experiments.

Number	Type of Network	Inputs	Outputs	Training Alg.	Hidden layer	Hidden neurons
1	Feed-Forward	10	10	Trainlm	1	10
2	Feed-Forward	10	10	Trainscg	1	10
3	Feed-Forward	10	10	Trainrp	1	10
4	Feed-Forward	10	10	Trainlm	1	20
5	Feed-Forward	10	10	Trainscg	1	20
6	Feed-Forward	10	10	Trainrp	1	20
7	Feed-Forward	10	10	Trainlm	2	10
8	Feed-Forward	10	10	Trainscg	2	10
9	Feed-Forward	10	10	Trainrp	2	10
10	Feed-Forward	10	10	Trainlm	2	20
11	Feed-Forward	10	10	Trainscg	2	20
12	Feed-Forward	10	10	Trainrp	2	20
13	Cascade-Forward	10	10	Trainlm	1	10
14	Cascade-Forward	10	10	Trainscg	1	10
15	Cascade-Forward	10	10	Trainrp	1	10
16	Cascade-Forward	10	10	Trainlm	1	20
17	Cascade-Forward	10	10	Trainscg	1	20
18	Cascade-Forward	10	10	Trainrp	1	20
19	Cascade-Forward	10	10	Trainlm	2	10
20	Cascade-Forward	10	10	Trainscg	2	10
21	Cascade-Forward	10	10	Trainrp	2	10
22	Cascade-Forward	10	10	Trainlm	2	20
23	Cascade-Forward	10	10	Trainscg	2	20
24	Cascade-Forward	10	10	Trainrp	2	20

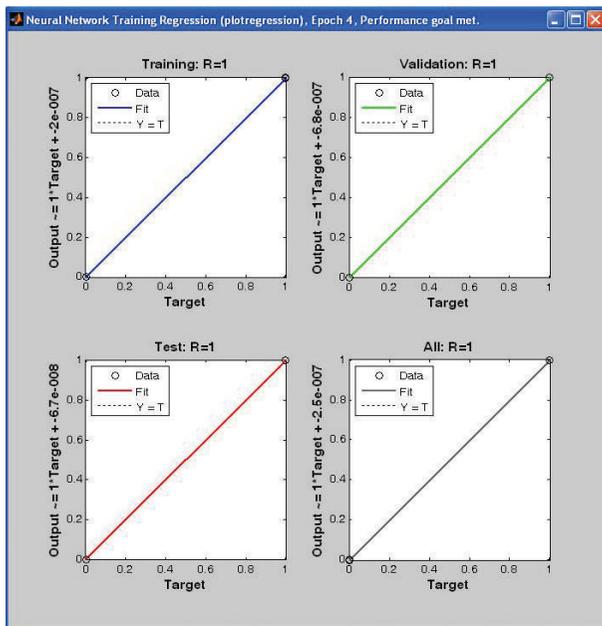


FIGURE 8: Regression plot for experiment no. 19.

perfect fit ($R = 1$). On the other hand, the validation result in experiment 1 shows low R value that is 0.77443 compared

to R value of experiment 19 which shows that the validation data fits in all iterations.

In order to choose between the two ANN models (no. 1 and no. 19), Table 4 shows the rate of correct diagnoses of each model in case of exposing them to a set of inputs with missing alarms and so to identify which model will perform best.

So it can be noticed that experiment 19 has highest rate of correct diagnoses, in spite of the wrong diagnoses but it was able to define the root cause of alarms in addition of some incorrect predicated links. So it was chosen as the best ANN model over all other models.

8. Ann Visualization

The inputs and the outputs of the ANN model were displayed on Google Earth application, so the BTS site is represented on the map as a yellow icon and the MW link is represented on the map as a yellow line in case of no alarm and it will be changed to red color in case of existing alarm on the site (BTS) or on the predicated failed link. Drawing the chosen area on Google Earth application was done by writing KML file and read by Google Earth application (see Figure 9).

The file is written by a toolbox added to the MATLAB program called "googleearth toolbox" and by default we assumed no alarms in the chosen area. In order to read the KML file which is written by the MATLAB program in a

TABLE 4: Rate of correct diagnoses.

ANN	Alarms					
	One missing Alarm		Less or equal two missing alarms		Less or equal three missing alarms	
	Correct	Wrong	Correct	Wrong	Correct	Wrong
Experiment 1	94.45%	5.55%	75.76%	24.24%	67.5%	32.5%
Experiment 19	94.45%	5.55%	78.79%	21.21%	70%	30%



FIGURE 9: ANN visualization in case of existing alarms.

continuous way, a KML code is written and opened by Google Earth application to read the KML file every defined time interval.

9. Conclusion

The presented work studied the feasibility of using ANN technique to solve the problem of alarm correlation in GSM networks. The presented ANN model was developed based on real GSM mobile network. The network was divided into different subareas; each area was represented by one neural network. The ANN model incorporated ten inputs; each input represented a BTS site and ten outputs and each output represented a MW link. Different combinations of training parameters have been carried out to choose the optimal network structure and configuration. Google Earth application was used to visualize the inputs and the outputs of the ANN model. The MATLAB program was used as a tool for development and implementation of the designed ANN model. It was observed that the ANN models demonstrated a good learning capability towards the training patterns presented. The selected ANN models showed a good performance progress and low *R* values in the regression plots.

Acknowledgment

The authors would like to express their gratefulness and appreciation to ALMADAR Company and its staff for their support throughout this research.

References

[1] S. Wallin, V. Leijon, and L. Landén, “Statistical analysis and prioritization of alarms in mobile networks,” *International*

Journal of Business Intelligence and Data Mining, vol. 4, no. 1, pp. 4–21, 2009.

[2] H. -C. Chao, “Modified adaptive resonance theory for alarm correlation based on distance hierarchy in mobile networks,” in *Proceedings of the 13th Asia-Pacific Network Operations and Management Symposium (APNOMS ’11)*, pp. 1–4, Taipei, Taiwan, 2011.

[3] D. S. Kim, H. Shinbo, and H. Yokota, “An alarm correlation algorithm for network management based on root cause analysis,” in *Proceedings of the 13th International Conference on Advanced Communication Technology (ICACT ’11)*, pp. 1233–1238, Seoul, Republic Of Korea, February 2011.

[4] T. Li and X. Li, “Novel alarm correlation analysis system based on association rules mining in telecommunication networks,” *Information Sciences*, vol. 180, no. 16, pp. 2960–2978, 2010.

[5] H. Wietgreffe, K.-D. Tuchs, K. Jobmann et al., “Using neural networks for alarm correlation in cellular phone networks,” in *Proceedings of the International Workshop on Applications of Neural Networks in Telecommunications*, pp. 1–10, 1997.

[6] H. Wietgreffe, “Investigation and practical assessment of alarm correlation methods for the use in GSM access networks,” in *Proceedings of the IEEE/IFIP Network Operations and Management Symposium*, pp. 391–403, Florence, Italy, April 2002.

[7] W. Jian and L. X. Ming, “A dynamic mining algorithm of association rules for alarm correlation in communication networks,” in *Proceedings of the 3rd IEEE/Create-Net International Conference on Communication System Software and Middleware (COMSWARE ’08)*, pp. 799–802, Bangalore, India, January 2008.

[8] O. Jukic and M. Kunštic, “ABCDE—Alarm Basic Correlations Discovery Environment,” in *Proceedings of the 33rd International Convention (MIPRO ’10)*, pp. 528–533, Opatija, Croatia, May 2010.

[9] M. Mouly and M.-B. Pautet, *The GSM System for Mobile Communications*, 1992.

[10] G. Jakobson and M. D. Weissmann, “Alarm correlation,” *IEEE Network Magazine*, pp. 52–59, 1993.

[11] H. Wietgreffe, “Performance modelling of network management traffic: an analytical approach,” in *Proceedings of the The 5th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI ’01)*, 2001.

[12] D. M. Meira and J. M. S. Nogueira, “A recursive approach for alarm correlation in telecommunication networks,” in *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS ’00)*, pp. 985–986, April 2000.



The Scientific World Journal

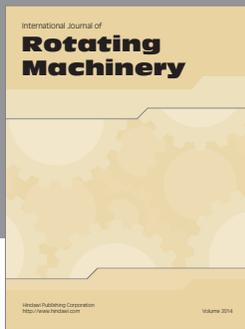
Hindawi Publishing Corporation
<http://www.hindawi.com>

Volume 2013



Hindawi

- ▶ Impact Factor **1.730**
- ▶ **28 Days** Fast Track Peer Review
- ▶ All Subject Areas of Science
- ▶ Submit at <http://www.tswj.com>



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

