

# A General Neural Network Model for Estimating Telecommunications Network Reliability

Fulya Altiparmak, Berna Dengiz, and Alice E. Smith, *Senior Member, IEEE*

**Abstract**—This paper puts forth a new encoding method for using neural network models to estimate the reliability of telecommunications networks with identical link reliabilities. Neural estimation is computationally speedy, and can be used during network design optimization by an iterative algorithm such as tabu search, or simulated annealing. Two significant drawbacks of previous approaches to using neural networks to model system reliability are the long vector length of the inputs required to represent the network link architecture, and the specificity of the neural network model to a certain system size. Our encoding method overcomes both of these drawbacks with a compact, general set of inputs that adequately describe the likely network reliability. We computationally demonstrate both the precision of the neural network estimate of reliability, and the ability of the neural network model to generalize to a variety of network sizes, including application to three actual large scale communications networks.

**Index Terms**—All-terminal network reliability, estimation, neural network.

## ACRONYM<sup>1</sup> & NOMENCLATURE

|         |   |
|---------|---|
| ANN     | artificial neural network   |
| BP      | ANN trained by back propagation of errors   |
| MAD     | mean absolute deviation   |
| ND      | node degree   |
| p-value | probability of occurrence of sample statistics if the compared results are $s$ -identical |
| RMSE    | root mean squared error   |
| t test  | two sample statistical test of means using Student t distribution                         |
| UB      | upper bound on network reliability  |

Manuscript received October 29, 2006; revised January 17, 2008 and September 15, 2008; accepted October 05, 2008. First published February 10, 2009; current version published March 04, 2009. This work was supported in part by the Scientific and Technical Research Council of Turkey (TUBITAK), the National Planning Center of Turkey (DPT) under Grant DPT-96K120820, and the U.S. National Science Foundation (NSF) under Grant INT-9731207. Associate Editor: M. Zuo.

F. Altiparmak is with Department of Industrial Engineering, Gazi University, Ankara, Turkey (e-mail: fulyaal@gazi.edu.tr).

B. Dengiz is with Department of Industrial Engineering, Baskent University, Ankara, Turkey (e-mail: dengiz@baskent.edu.tr).

A. E. Smith is with Department of Industrial and System Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: smithae@auburn.edu).

Digital Object Identifier 10.1109/TR.2008.2011854

<sup>1</sup>The singular and plural of an acronym are spelled the same.

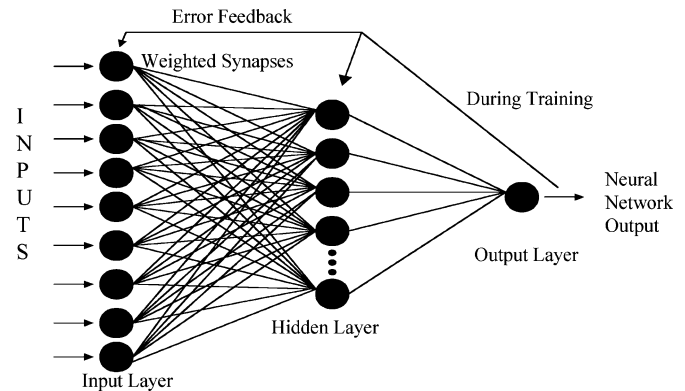


Fig. 1. Typical ANN components, and structure.

## I. INTRODUCTION

**I**N Studies on the design of communications networks, reliability has been defined in a number of ways. In this study, a probabilistic measure, all-terminal reliability, is considered (this is sometimes termed overall network reliability). All-terminal reliability is the probability that a set of operational edges provides communication paths between every pair of nodes. A communications network is typically modeled as a graph with  $N$  nodes, and  $L$  edges; nodes represent sites (computers), and edges represent communication links. Each node, and each edge has an associated probability of failure, and the reliability of the network is the probability that the network is operational. The definition of reliability thus depends on which components are operational.

In the literature, generally, researchers have made the following assumptions:

- i) Nodes are completely reliable; failure of links is the only cause of network failure.
- ii) Link failure probabilities are  $s$ -independent. This is consistent with the hypothesis that the agent causing failures is random [1], [2].
- iii) Link failures are equally probable. This assumption is often made because no detailed information about link failures is available, whereas information about the average failure is available [3].

There exists no algorithm with a polynomial time to compute all-terminal network reliability; therefore, the problem is NP-hard [4]. Although simulation is suitable for large networks, and is generally more flexible than analytical methods, it has disadvantages. The most notable is that it gives approximations; and when a high degree of accuracy is necessary, the running time to provide a desired confidence interval can grow large [5].



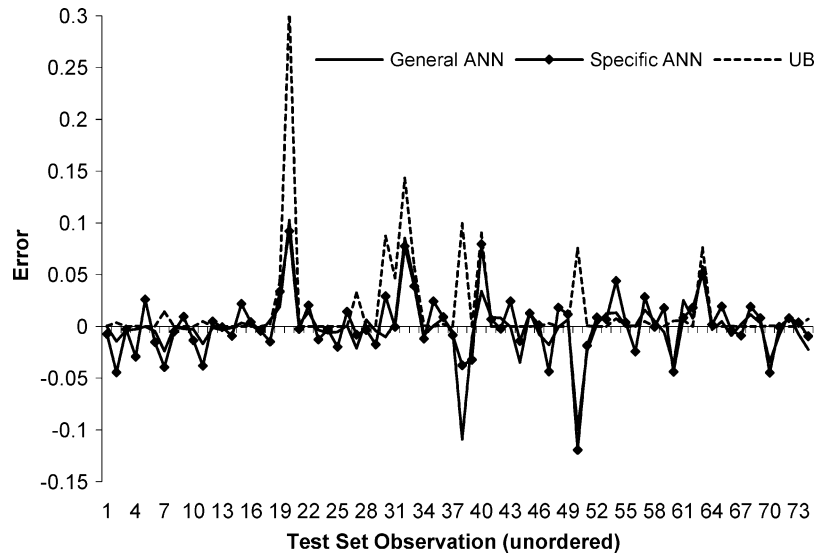


Fig. 3. Comparison of estimation error for networks with 20 nodes.

TABLE III  
RMSE, AND MAD VALUES OVER THE TEST SET FOR ALL MODELS

| Network  | General ANN |         | Specific ANN |         | Upper Bound |         |
|----------|-------------|---------|--------------|---------|-------------|---------|
|          | RMSE        | MAD     | RMSE         | MAD     | RMSE        | MAD     |
| 10 nodes | 0.01878     | 0.01374 | 0.02284      | 0.01555 | 0.03653     | 0.01870 |
| 15 nodes | 0.02356     | 0.02080 | 0.03042      | 0.02750 | 0.05634     | 0.04363 |
| 20 nodes | 0.02756     | 0.03300 | 0.03003      | 0.03503 | 0.04681     | 0.04915 |

approach for only highly reliable communications systems. Section V is the concluding discussion.

## II. ARTIFICIAL NEURAL NETWORKS

An ANN (see Fig. 1) has the ability to learn relationships between given sets of input and output data by changing the weights. This process is called: training the ANN. The most well known training algorithm is the Back Propagation (BP) algorithm [23], [24]. It minimizes the total sum of square error, which is the difference between the desired and actual output, using the gradient descent method. One of the most important properties of a trained ANN is its ability to generalize, which means that ANN can generate a satisfactory set of outputs from inputs that are not used during its training process [25].

The performance of the ANN model is a function of several design parameters such as the number of hidden layers, the number of hidden neurons in each hidden layer, the size of the training set, and the training parameters. Theoretical work in ANN has shown that a single hidden layer is sufficient to approximate any complex nonlinear function under quite general conditions [26], [27]. While too many hidden neurons can hinder the ANN's ability to generalize data not seen during training by causing over-fitting, too few hidden neurons can cripple its ability to learn the mapping at hand [28]. Experimentation, and validation are used to choose the proper hidden unit size.

One of the most difficult issues is that of validation. The technique of cross validation is particularly useful because it makes the most of a limited size data set. In this approach, the data set

TABLE IV  
RMSE, AND MAD VALUES FOR NETWORKS WITH NODES BETWEEN 10 AND 20

| Number of Nodes | General ANN |         | Upper Bound |         |
|-----------------|-------------|---------|-------------|---------|
|                 | RMSE        | MAD     | RMSE        | MAD     |
| 11              | 0.01844     | 0.01481 | 0.02870     | 0.01627 |
| 12              | 0.01663     | 0.01272 | 0.03216     | 0.01859 |
| 13              | 0.01960     | 0.01567 | 0.02666     | 0.01734 |
| 14              | 0.01485     | 0.01608 | 0.02821     | 0.02392 |
| 16              | 0.03007     | 0.02724 | 0.04432     | 0.03184 |
| 17              | 0.02294     | 0.02816 | 0.03768     | 0.03597 |
| 18              | 0.02293     | 0.02145 | 0.03635     | 0.03174 |
| 19              | 0.03619     | 0.03970 | 0.06473     | 0.09133 |

is divided (randomly) into multiple sets (cross validation would include  $n$  sets, where  $n$  is the data set size, while grouped cross validation would include  $< n$  sets). Network training takes part on all but one set, and that held-out set is used as a test set for that ANN. This is repeated for the number of sets chosen, each time using a different hold-out set. Errors are averaged over all tested ANN; this average error is used as an estimate of an ANN trained on all members of the data set, and then tested on the population of the domain being modeled [29].

## III. THE GENERAL ANN METHOD

We identified compact, easily calculated measures of network connectivity and reliability as the candidate set of inputs: ND (of each node, 0 if the node is not present), minimum node degree of the network ( $ND_{\min}$ ), median node degree of the network ( $ND_{\text{med}}$ ), maximum node degree of the network ( $ND_{\max}$ ), link reliability (LR), number of links (NL), link connectivity (C), and a network reliability upper bound (UB) (that of Jan [30]). Five input configurations were studied:

- 1) ND, LR, UB
- 2) ND, NL, LR, UB
- 3) ND, C, LR, UB
- 4) ND, NL, C, LR, UB

TABLE V  
FIVE FOLD CROSS VALIDATION RESULTS FOR LARGE NETWORKS

| Fold    | RMSE     |         |             |
|---------|----------|---------|-------------|
|         | Training | Testing | Upper Bound |
| 1       | 0.04183  | 0.04275 | 0.08124     |
| 2       | 0.04186  | 0.04370 | 0.07323     |
| 3       | 0.04252  | 0.04417 | 0.09483     |
| 4       | 0.04011  | 0.04049 | 0.08075     |
| 5       | 0.04309  | 0.04516 | 0.09517     |
| Average | 0.04188  | 0.04325 | 0.08504     |

TABLE VI  
RMSE, AND MAD VALUES FOR LARGE NETWORKS

| Number of Nodes | GNN-L   |         | Upper Bound |         |
|-----------------|---------|---------|-------------|---------|
|                 | RMSE    | MAD     | RMSE        | MAD     |
| 31              | 0.03847 | 0.04277 | 0.05014     | 0.08868 |
| 32              | 0.03770 | 0.03534 | 0.07563     | 0.06719 |
| 33              | 0.04388 | 0.05978 | 0.04975     | 0.07732 |
| 34              | 0.03974 | 0.03836 | 0.08916     | 0.09593 |
| 36              | 0.04341 | 0.03416 | 0.08328     | 0.07714 |
| 37              | 0.04561 | 0.05920 | 0.07286     | 0.09403 |
| 38              | 0.04364 | 0.04708 | 0.08723     | 0.07401 |
| 39              | 0.05533 | 0.05861 | 0.07823     | 0.05738 |

5)  $ND_{min}$ ,  $ND_{med}$ ,  $ND_{max}$ , NL, C, LR, UB.

We now show the method for a General ANN for design of networks from 10 to 20 nodes. Twenty input neurons are reserved for node degrees (to accommodate networks up to 20 nodes). For example, when data are sampled from a network with 10 nodes, node degrees are assigned to the first 10 input neurons, and the remaining 10 input neurons are set to zero. There are 22 input neurons for the first configuration, 23 for the second and third configurations, 24 for the fourth configuration, and 7 for the fifth configuration. While this topology representation uses up to 24 input neurons for a network with 20 nodes, the representation previously employed in the literature would use 190 input neurons just to represent the links for the same network. Fig. 2, and Table I show a network with 10 nodes, and the alternative encodings. The output of the ANN is the estimation of all-terminal network reliability (one real valued neuron). The target network reliability of each network is estimated using a Monte Carlo simulation method [31].

We used randomly generated data sets for training and validation considering five different link reliabilities (0.80, 0.85, 0.90, 0.95, and 0.99), and five different link connectivity values (1 to 5), so that there are 25 design points. An equal number of network topologies were generated for each design point. Networks of 10, 15, and 20 nodes were generated with  $L$  from  $N - 1$  (minimally connected) to  $N(N - 1)/2$  (fully connected) inclusive. (The labeling algorithm given in [32] is used to check network connectivity for each generated network.) After a preliminary experimental study, the number of hidden neurons, and training data size were set to 15, and 2400, respectively. The model was validated using five-fold cross validation [29], where each validation network was trained and tested using 2400, and 600 observations, respectively. A final application network was trained using all members of the data set, i.e. 3000 observations, and its

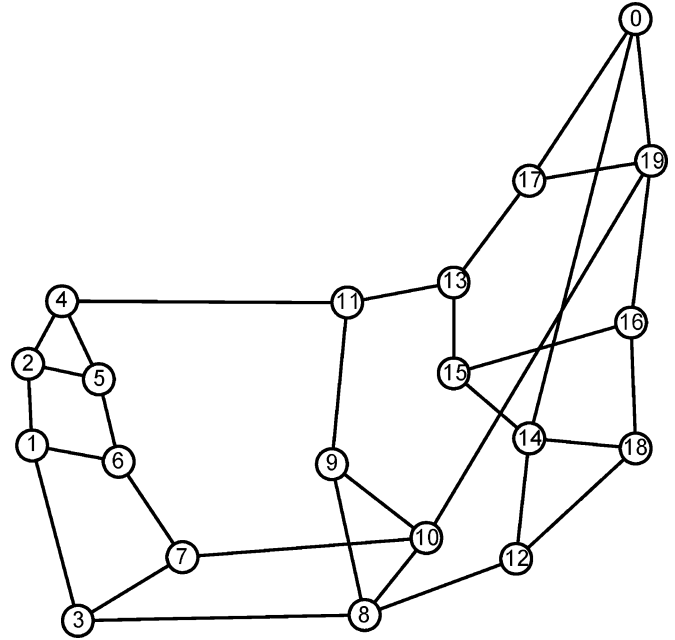


Fig. 4. Arpanet [33].

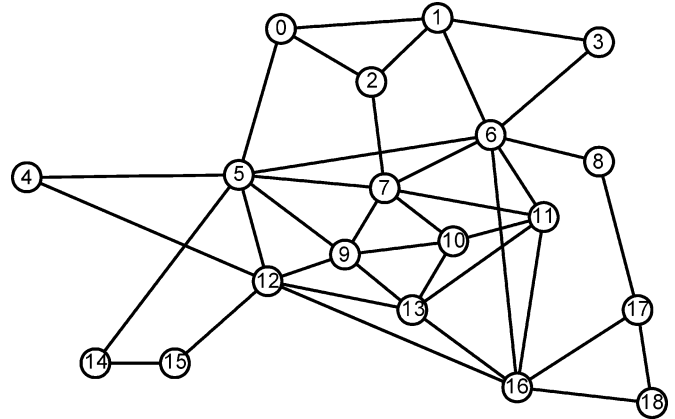


Fig. 5. European Optical Network [34], [35].

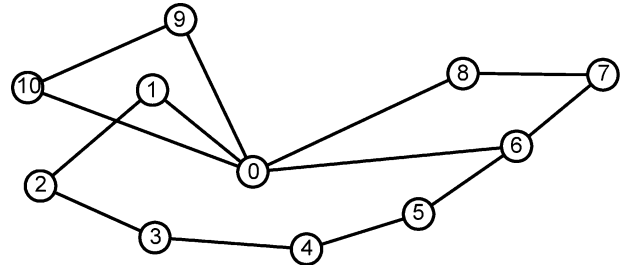


Fig. 6. Gazi University Network [6].

TABLE VII  
PROPERTIES OF THE REAL COMMUNICATIONS NETWORKS

| Name                  | Number of Nodes and Links |
|-----------------------|---------------------------|
| Arpanet               | $G = (20, 32)$            |
| European Optical Net  | $G = (19, 38)$            |
| Gazi Univ. Campus Net | $G = (11, 13)$            |

TABLE VIII  
RESULTS FOR ARPANET

| Link Reliability | Monte Carlo | Upper Bound | Specific ANN | General ANN ( $R(\mathbf{x}) \geq 0.90$ ) |
|------------------|-------------|-------------|--------------|---|
| 0.85             | 0.9192      | 0.9482      | 0.8908       | 0.9275                                    |
| 0.90             | 0.9765      | 0.9841      | 0.9367       | 0.9771                                    |
| 0.95             | 0.9972      | 0.9980      | 0.9648       | 0.9943                                    |
| 0.99             | 1           | 1           | 0.9804       | 0.9976                                    |

TABLE IX  
RESULTS FOR EUROPEAN OPTICAL NETWORK

| Link Reliability | Monte Carlo | Upper Bound | Specific ANN | General ANN ( $R(\mathbf{x}) \geq 0.90$ ) |
|------------------|-------------|-------------|--------------|---|
| 0.90             | 0.9307      | 0.9390      | 0.9478       | 0.9285                                    |
| 0.95             | 0.9848      | 0.9848      | 0.9875       | 0.9841                                    |
| 0.99             | 0.9994      | 0.9994      | 1.0050       | 0.9989                                    |

TABLE X  
RESULTS FOR GAZI UNIVERSITY NETWORK

| Link Reliability | Monte Carlo | Upper Bound | Specific ANN | General ANN ( $R(\mathbf{x}) \geq 0.90$ ) |
|------------------|-------------|-------------|--------------|---|
| 0.95             | 0.9525      | 0.9786      | 0.9243       | 0.9632                                    |
| 0.99             | 0.9978      | 0.9991      | 0.9646       | 0.9908                                    |

validation was inferred using the average of the prediction error of the five validation networks.

#### A. Comparison of Input Data Groupings

In this section, we give the results of five-fold cross validation for the neural networks considering the five different configurations. Five fold cross validation divides the data set randomly into five groups, then uses four to train the ANN, and one to test it; the procedure is repeated four more times, each with a different training group. Table II gives the  $s$ -average five-fold cross validation RMSE results for the five groupings. The error used to calculate RMSE is the difference between the Monte Carlo simulation, and the neural network estimation of the all-terminal network reliability. Ordering all input data configurations from the best to the worst according to their average RMSE values of the test set gives a sequence of 4, 2, 3, 1, 5. The UB RMSE column represents the RMSE of the UB only (no ANN estimation) for the testing set.

A statistical analysis using all 3000 observations shows that neural network estimations for input combination 4 were  $s$ -identical to the Monte Carlo estimation, while the upper bound was  $s$ -different from the Monte Carlo estimation. A paired t-test between the neural network and the Monte Carlo estimation has a p-value of 0.205 with a mean difference of  $6.49 \times 10^{-4}$ , while the same test between the UB and Monte Carlo estimation has a p-value  $< 0.0001$  with a mean difference of  $1.66 \times 10^{-2}$  for the same test.

#### B. Comparison of the General ANN, and Specific ANN Models

The performance of the chosen configuration (scheme 4) was further evaluated. The results compare network reliability estimations between this General ANN model, and ANN models specifically developed for each network size (10, 15, and 20 nodes). These latter networks were encoded in the traditional

way with binary link inputs, LR and UB (see Table I), and were trained on 1000 randomly generated instances of the specific size designated for that ANN.

The performance of the general model was compared with three specific models: ANN10, ANN15, and ANN20. For this comparison, a new test set containing 75 observations not used in training was generated randomly. Fig. 3 shows the estimation errors between the specific ANN, our General ANN, and the UB for a network size of 20 nodes. From this figure, it can be seen that all neural network estimations of reliability are unbiased, and very close to each other. Table III gives the performance measures (RMSE, and MAD) of all models. All RMSE values of the General ANN are smaller than the specific ANN, and the UB. This is surprising as the individual ANN are developed specifically for a certain network size. It appears that the information of summarizing aspects of a network architecture that affect all-terminal reliability (ND, NL, C) are used more efficiently by the neural network than a binary input of link presence or absence. (Both the general neural network, and the specific ANN include LR, and UB as inputs.)

#### C. The Performance of the General Model on New Network Sizes

We examined the performance of the General ANN for networks with node sizes from 10 to 20 not used in the training set (that is, other than 10, 15, and 20 nodes). The aim of this evaluation is to see how well our model can extend to network sizes unseen in training. The test set had 100 randomly generated instances of each network size. Table IV gives the RMSE, and MAD values for the General ANN, and the UB. We see no systematic error patterns in terms of network size. Therefore, it appears that the General ANN can be used to estimate all-terminal network reliability for any network size from 10 to 20 nodes with similar estimation error.

#### D. Scaling Up to Large Networks

To gauge scale up, we built & validated a General ANN in the same manner just described for networks from 30 through 40 nodes in size. As above, three different network sizes (30, 35, and 40 nodes) were used for training, and input configuration 4 was used. Table V gives five-fold cross validation results. While the average RMSE value is 0.04325 for the General ANN, it is 0.08504 for the upper bound. A paired t-test between the General ANN and Monte Carlo has a p-value of 0.1135 with a mean difference of 0.0011, while the paired t-test between the UB and Monte Carlo has a p-value  $< 0.00001$  with a mean difference of 0.0502 for the same test.

As before, we examined the performance of the General ANN for networks with node sizes from 30 to 40, other than the training node sizes of 30, 35, and 40. Table VI gives the RMSE, and MAD values. See that there are no systematic error patterns in terms of network size. These results show that the scale up of the General ANN approach is good, and that this approach is viable for networks of realistic size.

## IV. APPLICATION TO REAL COMMUNICATIONS SYSTEMS

We considered three real networks to better investigate the effectiveness of the General ANN approach. These are

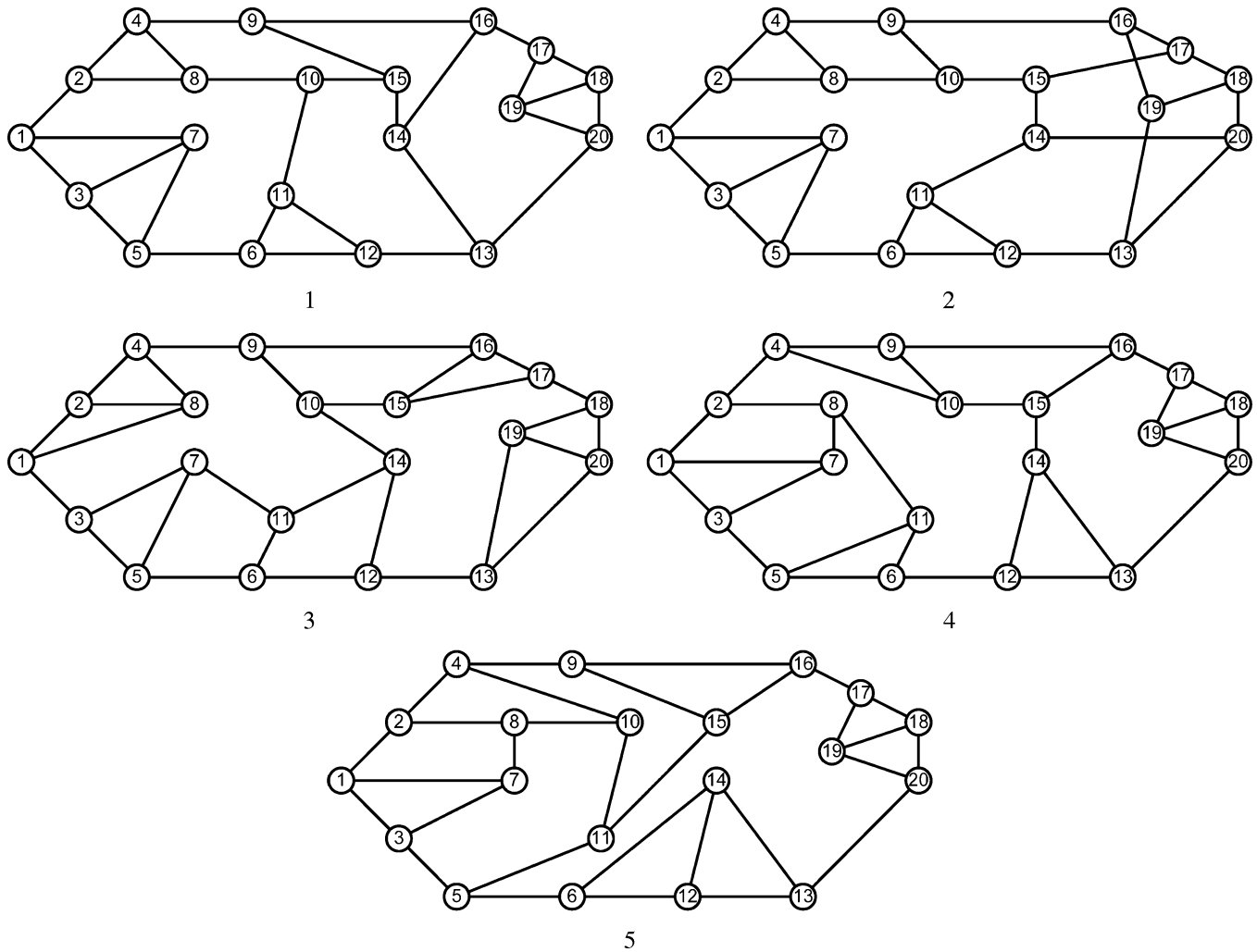


Fig. 7. Networks with ten 20 and 30 links nodes having same topological characteristics.

TABLE XI  
TOPOLOGICAL CHARACTERISTICS FOR SAMPLE NETWORKS WITH 20 NODES

|              |    |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |
|--------------|----|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
| Node         | 1  | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Degrees      | 3  | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  |
| Connectivity | 2  |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |
| Number of    | 30 |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |
| Links        |    |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |

Arpanet [33], the European Optical Network [34], [35], and the communications network of Gazi University [6], a large urban institution in Ankara Turkey with several campuses (see Figs. 4 through 6). The topological properties of the networks are shown in Table VII.

Because these communications networks are highly reliable, we trained & validated the ANN only on highly reliable networks, those with system reliability  $\geq 0.90$ . The General ANN estimates of system reliability were compared with the actual system reliability (using Monte Carlo simulation), the upper

bound of system reliability, and the estimate of system reliability using a specific ANN developed for that network architecture. Results are shown in Tables VIII through–X. The General ANN developed expressly for  $R(\mathbf{x}) \geq 0.90$  performed very well, better than both the UB, and the Specific ANN. In design optimization, one might use the General ANN for screening many designs to gauge the trade off between system reliability and cost. An exact method or computationally laborious Monte Carlo simulation should be used on the final few candidate network designs to ascertain the precise system reliability.

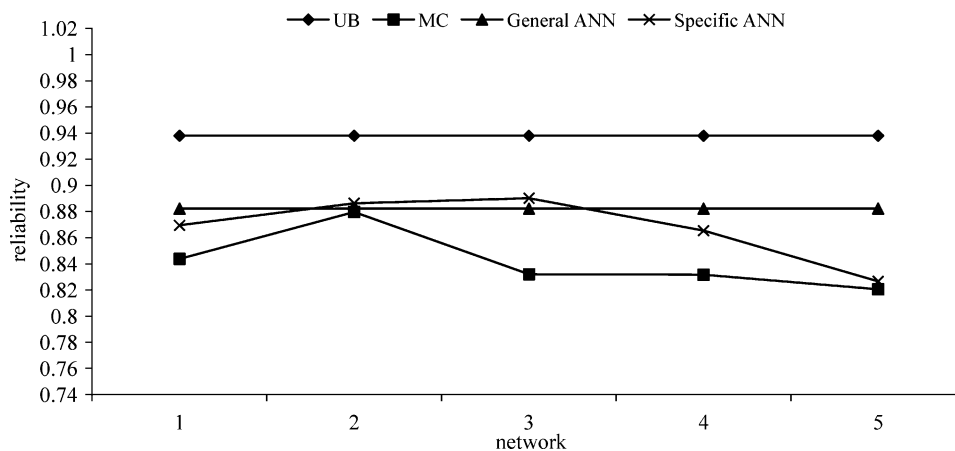


Fig. 8. Reliability estimations for the five networks of Fig. 7 from the four different methods.

## V. DISCUSSION

This paper presented a novel method of encoding communications networks for ANN that accommodates networks of varying node and link sizes. The inputs are compact (relatively few in number), and easily calculated. By using such an encoding, an ANN that is manageable in size, and flexible for many network design problems, can be trained & validated. This contrasts with previous work in neural network estimation of network reliability where the encoding was lengthy, and the resulting ANN could only be used for a single node size network.

Computational work shows that the General ANN is equivalent or superior in estimation accuracy to ANN models developed for a specific sized network. It was also shown that the General ANN could estimate network reliability with similar precision for sizes included in the training set, and not included in the training set, within the size domain considered (a range of ten nodes). Applying the method to larger, actual networks was successful.

We must note that there are different network topologies which yield the same values of the inputs to the General ANN. We studied this aspect by generating some differing networks with the same number of nodes, links, node degrees, link reliabilities, etc.; but which have different all terminal network reliability. The appendix gives one example of our studies. The General ANN approach cannot discriminate among networks with the same topological inputs. However, this does not invalidate using this approach during the screening of candidate designs, and then turning to the Monte Carlo approach for the few competitive designs for a more accurate reliability estimation.

## APPENDIX

Five different topologies of a 20 node/30 link network are shown in Fig. 7. Their topological characteristics are summarized in Table XI.

The reliabilities as estimated by the UB, Monte Carlo simulation (MC), the General ANN approach, and a specific ANN trained for that topology are shown in Fig. 8 for link reliabilities of 0.85. Of course, the General ANN gives the same estimation for all five networks; but it is more accurate than the UB for all networks, and similar to the specific ANN for several networks.

## REFERENCES

- [1] R. S. Wilkov, "Analysis and design of reliable computer networks," *IEEE Trans. Communication*, vol. COM-20, pp. 660–678, 1972.
- [2] R. M. Van Slyke and H. Frank, "Network reliability analysis I," *Networks*, vol. 1, pp. 279–290, 1972.
- [3] C. J. Colbourn and D. D. Harms, "Bounding all-terminal reliability in computer networks," *Networks*, vol. 18, pp. 1–12, 1988.
- [4] J. S. Provan and M. O. Ball, "The complexity of counting cuts and of computing the probability that a graph is connected," *SIAM Journals on Computing*, vol. 12, pp. 777–788, 1982.
- [5] M. Ball and R. M. Van Slyke, "Backtracking algorithms for network reliability analysis," *Annals of Discrete Mathematics*, vol. 1, pp. 49–64, 1977.
- [6] F. Altıparmak, B. Dengiz, and A. E. Smith, "Optimal design of reliable computer networks: A comparison of metaheuristics," *Journal of Heuristics*, vol. 9, pp. 471–487, 2003.
- [7] B. Dengiz, F. Altıparmak, and A. E. Smith, "Efficient optimization of all-terminal reliable networks, using an evolutionary approach," *IEEE Trans. Reliability*, vol. 46, pp. 18–26, 1997.
- [8] B. Dengiz, F. Altıparmak, and A. E. Smith, "Local search genetic algorithm for optimal design of reliable networks," *IEEE Trans. Evolutionary Computation*, vol. 1, pp. 179–188, 1997.
- [9] C. S. Cheng, Y. T. Hsu, and C. C. Wu, "Reliability evaluations of fault-tolerant systems based on neural networks," in *International Symposium on Artificial Neural Networks*, 1994, pp. 590–595.
- [10] C. S. Cheng, Y. T. Hsu, and C. C. Wu, "Fault modeling and reliability evaluations using artificial neural networks," in *IEEE International Conference on Neural Networks*, Perth, WA, 1995, pp. 427–432.
- [11] A. Mahmoud and M. Z. Monzoul, "Neural network for the reliability analysis of simplex systems," *Microelectronics and Reliability*, vol. 30, no. 4, pp. 795–800, 1990.
- [12] D. W. Coit and A. E. Smith, "Solving the redundancy allocation problem using a combined neural network/genetic algorithm approach," *Computers and Operations Research*, vol. 23, no. 6, pp. 515–526, 1996.
- [13] C. S. Cheng, Y. T. Hsu, and C. C. Wu, "An improved neural networks realization for reliability analysis," *Microelectronics on Reliability*, vol. 38, no. 3, pp. 345–352, 1998.
- [14] P. S. Rajpal, K. S. Shishodia, and G. S. Sekhon, "An artificial neural network for modeling reliability, availability and maintainability of a repairable system," *Reliability Engineering & System Safety*, vol. 91, pp. 809–819, 2006.
- [15] C. Srivaree-ratana, A. Konak, and A. E. Smith, "Estimation of All-terminal network reliability using an artificial neural network," *Computers and Operations Research*, vol. 29, pp. 849–868, 2002.
- [16] Z. Zhongding and L. Xiongjian, "A synthetic evaluation methodology based on neural networks theory for reliability indexes of communication networks," in *Proceedings of International Conference on Info-tech and Info-net*, 2001, vol. 2, ICII 2001, pp. 258–262.
- [17] A. Snow, P. Rastogi, and G. Weckman, "Assessing dependability of wireless networks using neural networks," in *Military Communications Conference, MILCOM 2005, IEEE*, 2005, vol. 5, pp. 2809–2815.

- [18] W. C. Yeh, C. Y. Yu, and C. H. Lin, "Evaluate voting system reliability using the Monte Carlo simulation and artificial neural network," in *2nd International Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2007)*, IEEE, 2007, pp. 45–49.
- [19] A. M. Leite da Silva, L. Chaves de Resende, L. A. Manso, and V. Miranda, "Composite reliability assessment based on Monte Carlo simulation and artificial neural networks," *IEEE Trans. Power Systems*, vol. 22, no. 3, pp. 1202–1209, 2007.
- [20] A. Dana, A. K. Zadeh, and S. A. S. Noori, "Backup path set selection in ad hoc wireless network using link expiration time," *Computers and Electrical Engineering*, vol. 34, no. 6, pp. 503–519, 2008.
- [21] E. Douglas, D. Torres, M. Claudio, and S. Rocco, "Reliability assessment of complex networks using rules extracted from trained ANN, and SVM models," in *Fifth International Conference on Hybrid Intelligent Systems, HIS'05*, 2005, pp. 1–6.
- [22] F. Altiparmak, B. Dengiz, and A. E. Smith, "Reliability estimation of computer communication networks: ANN models," in *IEEE ISCC'03 Conference Proceedings*, Antalya-Kemer, Turkey, 2003, pp. 1353–1358.
- [23] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Macmillan Publishing Company. New York: , 1994.
- [24] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representation by error propagation," in *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*. Cambridge, Massachusetts: MIT Press, 1986, vol. I, pp. 318–361.
- [25] L. K. Hansen and P. Solomon, "Neural network ensembles," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAM1-12, pp. 903–1002, 1990.
- [26] G. Cybenko, "Approximation by superposition of a sigmoidal function," *Mathematical Control Signals Systems*, vol. 2, pp. 303–314, 1989.
- [27] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1989.
- [28] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Computation*, vol. 4, pp. 1–58, 1992.
- [29] J. M. Twomey and A. E. Smith, "Bias and variance of validation methods for function optimization neural networks under conditions of sparse data," *IEEE Trans. System, Man and Cybernetics, Part C*, vol. 28, pp. 417–430, 1998.
- [30] R. H. Jan, "Design of reliable networks," *Computers and Operations Research*, vol. 20, no. 1, pp. 25–34, 1993.
- [31] M. S. Yeh, J. S. Lin, and W. C. Yeh, "A new Monte Carlo method for estimating network reliability," in *Proceedings of the 16th International Conference on Computers & Industrial Engineering*, 1994, pp. 723–726.
- [32] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows, Theory, Algorithms and Applications*. : Prentice Hall, 1993.
- [33] H. Choi, S. Subramanian, and H. A. Choi, "Lookback recovery from double-link failures in optical mesh networks," *IEEE/ACM Trans. Networking*, vol. 12, no. 6, pp. 1119–1130, 2004.
- [34] K. R. Venugopal, M. Shivakumar, and P. S. Kumar, "A heuristic for placement of limited range wavelength converters in all-optical networks," in *INFOCOM '99, 18th Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings*, 1999, vol. 2, pp. 908–915, IEEE.
- [35] X. Chu, B. Li, and Z. Zhang, "A dynamic RWA Algorithm in a wavelength-routed all-optical network with wavelength converters," in *INFOCOM 2003, 22nd Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings*, 2003, vol. 3, pp. 1795–1804.

**Fulya Altiparmak** is Professor of the Industrial Engineering Department at Gazi University. She has worked as researcher at the University of Pittsburgh in the USA, and visiting professor at the Waseda University in Japan. Her papers have been published in IEEE TRANSACTIONS ON RELIABILITY, IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, *Journal of Operational Research Society*, *Journal of Heuristics*, *Computers and Industrial Engineering*, and *OR Spectrum*.

**Berna Dengiz** is Professor and Dean of Engineering Faculty at Baskent University. She has received research funding for her collaborative studies from the NATO-B2 program, TUBITAK (The Scientific and Technical Research Council of Turkey), Government Planning Center of Turkey, and the National Science Foundation (NSF) of the USA. She has worked as visiting professor at the University of Pittsburgh, and the Auburn Universities. Her papers have appeared in IEEE TRANSACTIONS ON RELIABILITY, IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, *Journal of Operational Research Society*, *Journal of Heuristics*, *International Journal of Production Economics*, and *Simulation Modeling Practice and Theory*.

**Alice E. Smith** (SM'96) is Professor and Chair of the Industrial and Systems Engineering Department at Auburn University. She has served as a principal investigator on dollar4 million of sponsored research, holds one U.S. patent, and has authored more than 60 publications in journals which have garnered over 850 citations (Web of Science). Dr. Smith holds editorial positions on *INFORMS Journal on Computing*, *Computers & Operations Research*, *International Journal of General Systems*, IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, and *IIE Transactions*. Dr. Smith is a fellow of IIE, a senior member of IEEE, and a Registered Professional Engineer.