MODULAR ARTIFICIAL NEURAL NETWORKS FOR SOLVING THE INVERSE TRANSPORTATION PLANNING PROBLEM

Adel W. Sadek, Ph.D.*
Assistant Professor
Department of Civil and Environmental Engineering
University of Vermont
Burlington, VT 05405
Phone: (802) 656-4126
FAX: (802) 656-8446
E-mail: asadek@emba.uvm.edu

&

Charles Mark
Research Assistant
Department of Civil and Environmental Engineering
University of Vermont
Burlington, VT 05405
Phone: (802) 656-8252
FAX: (802) 656-8446
E-mail: cmark@emba.uvm.edu

Transportation Research Board
82nd Annual Meeting
Washington, D.C.

* Corresponding Author

Word Count: 4,823 words + (1 Table + 9 Figures) = 7,323 equivalent words
ABSTRACT

Given that it is extremely unlikely that the coming years will witness major capacity-expansion projects, transportation planners will now need to view the existing infrastructure as fixed, and start thinking about how much development the current system can sustain. This line of thinking, which involves deriving land-use limits from infrastructure capacity, requires solving the inverse of the typical transportation-planning problem. In this study, modular Artificial Neural Networks (ANNs) are developed for solving the inverse transportation planning problem. The ANNs are designed to predict zonal trip ends given the traffic volumes on the links of the transportation network. Computational experiments are performed in order to study the effect on the ANN accuracy of the following three factors: (1) the transportation network size; (2) the variability in the training data; and (3) the ANN topology. The study shows that ANNs are quite capable of capturing the relationship between link volumes and zonal trip ends for both small as well as medium-sized transportation networks, and for different degrees of variability in the training data. The study also shows that modular ANNs with one or two hidden layers appear to outperform other ANN topologies.

Key Words: Neural Networks, Computational Intelligence, Four-step Planning Process
MODULAR ARTIFICIAL NEURAL NETWORKS FOR SOLVING THE INVERSE TRANSPORTATION PLANNING PROBLEM

INTRODUCTION

With the continued growth in traffic volumes, transportation professionals are faced with the challenge of how to best alleviate the nation’s congestion problems. Historically, congestion problems were addressed using both supply-side and demand-side initiatives. Supply-side initiatives typically included activities such as infrastructure expansion and, more recently, Intelligent Transportation Systems (ITS) projects. Demand-side initiatives, on the other hand, involve programs such as employer-based travel demand management and telecommuting. However, while these strategies have had some success in alleviating congestion, the continued growth is clearly pointing to the need for a more drastic solution to the problem.

Recently, there has been a renewed interest in better understanding and redesigning the land use-transportation system (LUTS), in an attempt to fight congestion more effectively. This renewed interest in the LUTS connection is not only motivated by the need to relieve congestion, but, more importantly, by the increased national interest in environmental protection and sustainability. The idea here is to attempt to alleviate congestion through a better design of the land-use system, which could help avoid or reduce the need for major capacity-expansion projects. In other words, this approach assumes that the existing infrastructure is more or less fixed, and attempts to determine how much development the existing system can sustain.

This new line of thinking, which involves attempting to derive land-use limits as a function of the existing capacity, requires one to solve the inverse of the typical transportation-planning problem. In the typical transportation-planning problem, one starts with a given land use pattern, and then predicts future traffic volumes on the transportation system. However, to derive land-use limits, one needs to reverse the direction of the transportation planning process; that is to say, one needs to start with the transportation system characteristics and link volumes, and use these to derive land use limits in terms of the number of zonal trips generated from and attracted to each zone. Once the numbers of trips generated from and attracted to a given zone are determined, deriving land use limits for that zone becomes possible, based on knowledge of the trip generation characteristics of the different land use types.

Given this, we define the inverse transportation planning problem in this paper as follows. Given a transportation network and a desirable traffic pattern, we need to determine the maximum number of trips attracted to and generated from each zone that would ensure that what the desirable traffic pattern is attainable (by a desirable pattern, we mean traffic volumes that are well below the capacity of each link – for example, volumes corresponding to a Level of Service C). It should be noted, however, that solving the inverse of the transportation planning process is different from the so-called “integrated land-use transport (ILUT) models”, which were studied by the International Study Group on Land-Use Transport Interaction (ISGLUTI) under the auspices of the U.K. Transport and Road Research Laboratory (1). ILUT models essentially involve linking transportation planning models to land-use models, such as EMPAL and DRAM, to allow for two-way linkages from the land use to the travel demand models and vice versa. The problem is also different from the problem of deducing the Origin-Destination (O-D) matrix.
from the link volumes; a problem which has received a lot of attention in the transportation literature, and which still remains a challenging problem. The problem we are considering here is concerned with determining the total trip ends, and not with estimating the individual entries of the O-D matrix. In addition, the O-D problem is typically concerned with determining the current O-D matrix that corresponds to the current traffic pattern. Solving the inverse transportation planning problem, on the other hand, would allow for determining land use limits that would result in a future desirable traffic pattern.

Previous Attempts

To the authors’ knowledge, Miller and Demetsky were the first to attempt to solve the inverse transportation planning problem, in the sense the problem is defined in the current paper (2). Miller and Demetsky’s initial efforts to address the problem focused on attempting to solve the inverse of each of the steps in the famous 4-step planning process. Their goal was first to deduce the O-D matrix from the link volumes (a problem which, as previously mentioned, has received a lot of attention recently) and then to deduce the zonal trip ends from the deduced O-D matrix. However, according to the researchers, this approach performed very poorly to be of use, particularly because of the complexity of deriving the O-D matrix from traffic counts (3). As an alternative, Miller and Demetsky opted for the use of a very simple, direct estimation procedure. This procedure is based on using regression analysis to develop a model that directly estimates zonal trip ends as a function of factors such as link volumes, roadway types, travel speeds, and the location of a zone relative to the other zones in the region. The Mean Absolute Error (MAE) for the model, however, was rather significant.

In a more recent study, we have investigated the transferability of the relations proposed by Miller and Demetsky, using real-world data from the State of Vermont. Our results, however, show that the transferability of the models is somewhat questionable (4). We have also explored the feasibility of using Artificial Neural Networks (ANN) to solve the inverse transportation planning problem. For this purpose, we have trained a simple Multi-layer Perceptron (MLP) neural network to solve the inverse problem for a small real-world network consisting of 14 zones and 116 links. Our preliminary results clearly demonstrate the feasibility of the ANN approach.

PURPOSE AND SCOPE

The purpose of the current paper is to build on our initial efforts in using ANN to solve the inverse transportation planning problem. Specifically, in this paper, we explore additional network architectures and topologies (such as Modular ANN), and consider larger transportation networks compared to what we considered in our previous study. We also investigate the effect of the transportation network size and variability in the training data set, on the ANN performance and results.

The current paper is organized as follows. First, some background information on Modular ANN and how they compare to the famous MLP network is provided. This is followed by a description of the test networks used in the study, and a discussion of how the data needed to train the ANNs were generated. Next, the computational experiments performed and the results
obtained are discussed. The paper concludes by summarizing the major conclusions derived from the study.

ARTIFICIAL NEURAL NETWORKS

ANNs are biologically-inspired systems consisting of a massively connected network of computational “neurons”, organized in layers. By adjusting the weights of the network, ANNs can be “trained” to approximate virtually any nonlinear function to a required degree of accuracy. ANNs typically learn by providing the network with a set of input and output exemplars (5). A learning algorithm (such as back propagation) would then be used to adjust the weights of the network so that the network would give the desired output, in a type of learning commonly called supervised learning. Over the years, several ANNs types and architectures were developed, among the most important of which is the Multi-layer Perceptron (MLP) Neural Network. The MLP typically consists of three layers: the input layer, the hidden layer(s), and the output layer as shown in Figure 1. The type of connections in the MLP is of the feedforward type, where all possible connections between the different neurons are made.

Modular Feedforward Networks

As the name indicates, Modular Feedforward Networks are special cases of MLPs, in which the hidden layer(s) are segmented into modules. This segmentation helps create some structure within the network topology, and allows for the specialization of the function within each submodule. As opposed to MLP, Modular Feedforward Networks do not have full interconnectivity between the layers, resulting in a smaller number of weights compared to a MLP of the same size, especially when more than one hidden layer is used (Figure 2). This in turn tends to speed training. It also helps reduce the number of examples needed to train the network to the same degree of accuracy, since the number of examples required is a direct function of the number of weights in the network.

The inverse transportation problem we are considering in this study involves predicting the number of trip ends for several zones and not just a single zone. Given this, the use of a Modular ANN seems appropriate, since this will allow for defining some structure within the ANN so that each sub-network (within the Modular network) would be responsible for predicting the trip ends for a sub-set of the zones. The challenge in using Modular Networks, however, lies in the fact that there are typically many ways to segment a MLP into modules. Moreover, it is usually unclear how to best design a modular topology (6). Given this, it is important to experiment with different topologies and to select the most appropriate for the problem at hand.

Rationale for Using an ANN to Solve the Inverse Transportation Problem

One approach to solving an inverse problem is to directly solve for the inverse if the exact function is known. It is relatively easy to find the inverse function for a one-variable function that exhibits one-to-one behavior (each input results in a unique output). However, for the inverse transportation planning problem, we do not know the exact function that relates link volume to zonal trip ends. Even if the function were known, it would involve multiple variables
and would definitely not be one-to-one since it is possible for different zonal trip ends to result in the same link volumes.

ANNs are a good computational tool to use to solve the inverse transportation problem since they are capable of relating a system’s inputs to its outputs with no knowledge of how the system specifically functions. Instead of relying on a deductive approach (understanding a system’s behavior to create a general model used to evaluate a specific scenario), ANNs take advantage of inductive approach (observing many system responses to various changes in input parameters to create a general model that can evaluate specific scenarios.)

In addition to the above, the use of ANNs to solve the inverse transportation problem offers another advantage. As previously mentioned, the inverse transportation problem does not exhibit one-to-one behavior, which means that it is possible for multiple zonal trip end values to result in the same link volumes. With the use of an ANN, however, the structure of the predicted land use pattern (in terms of the relative values of the trips ends for the different zones) would be similar to the structure of the land use pattern utilized in generating the data set used to train the ANN. This would make the ANN solution quite realistic, since, as will be explained later in this paper, the training data set will be generated in such a fashion so as to preserve the structure of the original land use pattern to a certain extent.

**TRANSPORTATION TEST NETWORKS**

In order to experiment with different ANN topologies for solving the inverse transportation planning process, a study area was needed. For this purpose, we selected the Chittenden County region in Northwestern Vermont. Since the late 1980s, the Chittenden County Metropolitan Planning Organization (CCMPO) has maintained an excellent transportation model for the region (Figure 3). From this model, two subnetworks were extracted for the purposes of developing the ANNs and for testing the effect of the transportation network size on the ANN accuracy. The sub-networks were selected from the larger network because each represented a compact independent system. Political and geographical boundaries were used as guides to identify potential sub-networks.

The first subnetwork was the network corresponding to the City of Winooski, Vermont. The Winooski network had a total of 14 zones, 45 nodes, and 114 links, and hence represented a small-sized transportation network (we will refer to this transportation network in the paper as the “small” network). The second subnetwork extracted was that corresponding to the town of Essex Junction, with a total of 34 zones, 115 nodes, and 298 links. The Essex Junction network represented a medium-sized area, and will henceforth be referred to as the “medium” network.

**Training Data Generation**

In order to generate the data required for training the ANNs, the first step was to generate a large number of O-D matrices by scaling the current O-D matrix for the Winooski’s network, and the Essex Junction’s network. To do this, a simple program was written in MATLAB. The program took the current O/D matrix and first scaled it by multiplying it by a random factor in the range
of (0 – 1.0). The program then added noise to the value of each cell in the O/D matrix in order to cover the solution space of the problem. Without adding the noise, the relative values of the trip interchanges (i.e. the number of trips between a given O-D pair) relative to one another would remain unchanged from the original O-D matrix. Although this would artificially make solving the inverse of the transportation problem easier, it is not desirable since we would like the ANN model to be able to solve the inverse problem when there is indeed a change in the relative values of the trip interchanges of the O/D matrix.

In adding noise to the O/D matrix entries, two noise values were used (namely ± 20% and ± 40% of the cell entry), in order to study the effect of data variability on the ANN performance. This resulted in two training data sets for each transportation network. The 20% noise data set would represent the case when the structure (i.e. the relative values of the O/D trip interchanges) is modestly different from the original O/D matrix, whereas the 40% noise data set would represent the case when there is a significant change in the O/D matrix structure. Using the MATLAB program, a total of 1000 O/D matrices were generated for the “small” transportation network (the Winooski network), and a total of 10,000 matrices for the “medium” transportation network (the Essex Junction network). For each of these O/D matrices and each transportation network, the transportation planning software (TP-Plus) was then used to assign the matrix to the corresponding network, and the resulting link volumes were recorded. This data was used to generate the input-output exemplars for the ANNs as explained below.

To model the inverse of the transportation-planning problem (i.e. the problem of predicting trip ends from link volumes), the input to the ANNs would have to represent the link volumes, and the desired output would have to correspond to the number of trip ends from each zone. To prepare the required data, therefore, the zonal trip ends (the desired output) were first calculated by summing the corresponding rows and columns from the O-D matrix. For the input data, the first step was to exclude those links that corresponded to the zones’ connectors, from the list of the link volumes that would be used as input to the ANN. Connectors are hypothetical links added to a transportation model in order to connect the zones to the network; they thus essentially represent the local street network in a given zone. Traffic volumes on the set of the connectors had to be excluded, because including them in the input data would make the problem of predicting trip ends from link volumes trivial (essentially the number of trip ends for a given zone is equal to the total volume on the connectors joining that zone to the network). Also, given the fact that the connectors are hypothetical links that do not correspond to real links in the real-world transportation system, one should not expect traffic volumes to be available for these links.

After excluding the connectors and unimportant links from the network, we ended up with a total of 46 links for the Winooski network, and a total of 179 links for the Essex Junction network. As mentioned above, traffic volumes on the links would constitute the input to the ANN models. That is to say, an input-output exemplar to the ANN for the Winooski network (i.e. the “small” network) consisted of a 46-element input vector (giving the traffic volume on each of the 46 links), and the corresponding 14-element output vector, giving the total trip ends for the 14 zones of the test network. For the Essex Junction network, the input-output exemplar was in the form of a 179-element input vector giving the traffic volume on the 179 links, and the corresponding 34-element vector, giving the total trip ends for the 34 zones.
COMPUTATIONAL EXPERIMENTS AND RESULTS

In this study, we have experimented with a number of Modular ANN models for modeling the relationship between link volumes, and the zonal trip ends (the inverse transportation planning problem). The first topology we tried was a modular ANN network with one hidden layer. For the Winooski transportation network, the ANN had 46 neurons in the input layer (corresponding to the 46 transportation network links); 14 neurons in the hidden layer, divided into two submodules (see Figure 2 above); and 14 neurons in the output layer giving the zonal trip ends for each of the 14 zones in the Winooski network. The ANN developed for the Essex Junction network, on the other hand, had 179 neurons in the input layer; 34 neurons in the hidden layer, divided into two submodules; and 34 neurons in the output layer.

The ANN models were developed using NeuroSolutions software from NeuroDimension, Inc (6). The training data set was divided into three groups: the training data set (70%), the cross-validation data set (20%), and the testing data set (10%). The cross-validation set was used to test the ability of a network to generalize, while the network was still being trained. This helps safeguard against the possibility of the network “memorizing” the training pattern (over-fitting), which could lead to deterioration in the ability of the network to generalize. The back-propagation algorithm was used to train the networks, and training was continued until there was no further improvement in the mean square error for the cross-validation data set during 100 epochs, or until the number of epochs reached 1000. A third data set (the testing data set) was then used to test the accuracy of the networks after training.

Effect of Transportation Network Size on ANN Performance

Using the 20% noise data set, Modular ANNs were developed for both the small and medium transportation networks as described above. In both cases, the training sessions exhausted the 1000 epochs before the cross validations error started to increase. However, the mean square error (MSE) became stable and was converging to zero, which is a strong indication that the ANN’s training was successful.

After the ANNs were trained, the weights were frozen and a set of data the network had not trained on (i.e. the test data set) was run through the network. The outputs from both ANNs were compared to the actual values. The percent error between the actual and calculated data for both networks was calculated using the formula:

\[
\% \text{ error} = \frac{\sqrt{(actual - calculated)^2}}{actual}
\]  

(Eq. 1)

The average percent error, averaged over the test set, for the ANN developed for the small transportation network (Winooski) was 3.61%. This compares to a value of 3.11% for the ANN developed for the medium transportation network (Essex Junction). The similar average errors incurred for the small and medium transportation networks suggest that this procedure is scalable to larger, more complex transportation networks and not limited to smaller, simpler transportation networks.
The level of error that occurred is quite acceptable when the sensitivity required in this application (transportation planning) is considered. For example, Table 1 shows a sample of actual values for the zonal trip ends against values calculated by the ANN for the small and medium transportation networks. Some of the samples exhibited error under 1%, which produced highly accurate approximations. But even samples with errors above 5% were able to reasonably approximate the actual value. The practical difference between an O/D value of 886 and 924 will most likely have an insignificant effect in a transportation planning context.

Figures 4 and 5 plot the actual values against the calculated ANN values. The 45° line represents the location where the data would be plotted if the calculated values were exactly equal to the actual values. The closer the data is plotted to the line, the better the calculated data approximates the actual data. Both Figures 4 and 5 have data points that fit closely to the 45° degree line. This is further proof that the ANN was able to successfully derive the zonal trip ends from link volumes for the network sizes examined in this study.

**Effect of Variability in the O/D Matrix Structure on ANN Performance**

In order to study the effect of variability in the O/D matrix structure on the ANN performance, the study considered the medium transportation network (the Essex Junction network), and trained two modular ANNs, one using the training data set where the noise introduced was within ±20%, and another using the ±40% noise data set. The procedure to train and test the ANN was as described above.

For the case of ±40% noise, the average percent error for the test set was 5.26%. This compares to the previous value of 3.11% obtained using the ±20% noise data set. As expected, introducing more variability into the O/D matrix structure produced higher errors. Nevertheless, a 5.26% average error is still quite acceptable for transportation planning applications. Figure 6 plots the actual versus the calculated zonal trip ends for the ANN trained using the ±40% noise data set. As can be seen, more dispersion about the 45° line can be detected in this case, compared to the ±20% noise data set case (Figure 5). Nevertheless, the accuracy level appears to be quite acceptable.

**Effect of ANN Topology**

In this study, we experimented with a number of different ANN topologies in order to study the effect of the ANN topology on its performance. Figure 7 shows the four different topologies tested: (A) a multi-layer perception (MLP), (B) a modular ANN with 1 hidden layer, (C) a modular ANN with 2 hidden layers and connections to the output layer after the first hidden layer, and (D) a modular ANN with 1 hidden layer and a direct connection from the input to the output layer. In testing these different topologies, we used the same medium-sized transportation network and the considered ±40% noise data set, since this was the case that yielded the largest average percent error. The training and testing of each of the four ANNs were conducted in a fashion similar to that previously discussed.

Figure 8 displays the actual zonal trip ends versus the values predicted by the 4 ANN topographies for each of the 34 zones for one set of testing data. The bars represent the value of
the actual data while the individual points are the values predicted by the 4 ANN topologies. As can be seen, while all ANNs appear to be capable of capturing the relationship between link volumes and zonal trip ends, topology (B) appears to be the best topology for capturing this relationship.

To more accurately examine the effect of the ANN topology on performance, the mean error for the four topologies A, B, C, and D was calculated. This was done by first finding the mean error for 1000 test data sets for each of the 34 zones. Then the mean errors for each zone were averaged together to produce values of 6.31%, 5.26%, 6.02%, and 8.82% for the mean errors for topologies A, B, C and D, respectively. Figure 9 plots the 95% confidence intervals for the four topologies’ errors. As can be seen from the figure, topology D’s error appears to be higher than the other three errors. Statistical tests also revealed that there are statistical differences among the four means, when pairwise comparisons between the means were conducted.

Given this, it seems that while all ANN topologies appear to yield an acceptable level of error, topology (B) appears to yield the best performance, and topology (D) appears to yield the worst. The slightly inferior performance of topology D compared to the other topologies could be attributed to the fact that, in topology (D), some data by-passed the hidden layers and was sent directly to the output layer, which uses a linear manipulation function. The data that by-passed the hidden layer was never subject to any non-linear manipulation, resulting in a slightly higher error value. Since the performance of the four topologies appears to be acceptable, other factors may be considered when deciding which topology to implement. These factors include the ability of the learning curve to converge with less noise, the number of data points required to train a network, and the time needed to train the network.

CONCLUSIONS

In this study, modular ANNs were developed for solving the inverse transportation planning problem. The ANNs were designed to predict zonal trip ends given the traffic volumes on the links of the transportation network. Computational experiments were performed in order to study the effect on the ANN accuracy of the following three factors: (1) the transportation network size; (2) the variability in the training data; and (3) the ANN topology. It should be noted that the proposed procedure, as outlined, is technically not solving the inverse planning process directly, but rather solving the inverse of the TP-Plus’s (i.e. the transportation planning software) simulations of the transportation process. The ANN solution, however, would still provide valuable insights into the land use pattern that would help prevent congestion on the transportation network, provided that one can make the assumption that the four-step planning process, as coded in the TP-Plus model, closely represents real-world conditions.

Among the main conclusions of the study are:

(1) Modular ANNs are quite capable of mapping the relationship between network link volumes and zonal trip ends, and hence can be used to predict land-use limits from infrastructure capacity. The training data can be easily generated from an existing transportation planning model for the region.
(2) ANNs appear to be able to successfully derive zonal trip ends from link volumes, regardless of the transportation network size (i.e. number of zones) and complexity.

(3) ANNs are capable of capturing the relationship between link volumes and zonal trip ends, even when the structure of the O/D matrix in terms of the relative values of the O/D trip interchanges is significantly different from the original O/D matrix.

(4) A modular ANN topology with one or two hidden layers appears to outperform other ANN topologies (such as MLP) in predicting zonal trip ends from link volumes.

ACKNOWLEDGEMENTS

This research is being funded by a grant from the National Science Foundation (NSF). The authors would like to thank NSF for its support.
REFERENCES

LIST OF TABLES

TABLE 1. Sample Actual and Calculated Values for the Small and Medium Networks
LIST OF FIGURES

Figure 1. The Multi-Layer Perceptron (MLP) Network

Figure 2. Modular Artificial Neural Networks

Figure 3. Chittenden County Transportation Planning Model

Figure 4. Actual vs. Predicted Zonal Trip Ends for the Winooski Network and the ± 20% Noise Data Set

Figure 5. Actual vs. Predicted Zonal Trip Ends for the Essex Junction Network and the ± 20% Noise Data Set

Figure 6. Actual vs. Predicted Zonal Trip Ends for the Essex Junction Network and the ± 40% Noise Data Set

Figure 7. The Four ANN Topologies

Figure 8. Comparison of the Performance of the Four ANN Topologies

Figure 9. 95% Confidence Intervals for the % Error of the Four Network Topologies
Table 1. Sample Actual and Calculated Values for the Small and Medium Networks

<table>
<thead>
<tr>
<th>Small Transportation Network</th>
<th>Medium Transportation Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
<td>Calculated</td>
</tr>
<tr>
<td>572.00</td>
<td>559.31</td>
</tr>
<tr>
<td>683.00</td>
<td>683.17</td>
</tr>
<tr>
<td>900.00</td>
<td>907.51</td>
</tr>
<tr>
<td>950.00</td>
<td>995.70</td>
</tr>
<tr>
<td>1001.00</td>
<td>1051.77</td>
</tr>
<tr>
<td>1075.00</td>
<td>1057.73</td>
</tr>
<tr>
<td>1253.00</td>
<td>1256.11</td>
</tr>
<tr>
<td>1331.00</td>
<td>1316.94</td>
</tr>
<tr>
<td>1531.00</td>
<td>1535.56</td>
</tr>
<tr>
<td>1562.00</td>
<td>1474.76</td>
</tr>
</tbody>
</table>
Figure 1. The Multi-Layer Perceptron (MLP) Network
Figure 2. Modular Artificial Neural Networks
Figure 3. Chittenden County Transportation Planning Model

Chittenden County Transportation Model
Figure 4. Actual vs. Predicted Zonal Trip Ends for the Winooski Network and the ± 20% Noise Data Set
Figure 5. Actual vs. Predicted Zonal Trip Ends for the Essex Junction Network and the ± 20% Noise Data Set
Figure 6. Actual vs. Predicted Zonal Trip Ends for the Essex Junction Network and the ± 40% Noise Data Set
Figure 7. The Four ANN Topologies
Figure 8. Comparison of the Performance of the Four ANN Topologies
Figure 9. 95% Confidence Intervals for the % Error of the Four Network Topologies