Performance Analysis of Cellular Radio System Using Artificial Neural Networks

Kriti Priya Gupta¹ *, Madhu Jain²

¹Symbiosis Centre for Management Studies, NOIDA Faculty of Management, Symbiosis International University, Pune, India
²Department of Mathematics, Indian Institute of Technology (IIT), Roorkee, India

Email address:
kriti.gupta@scmsnoida.ac.in (K. P. Gupta), madhujain@sancharnet.in (M. Jain)

To cite this article:

Received: December 26, 2016; Accepted: January 6, 2017; Published: March 17, 2017

Abstract: In this paper, we exploit one of the fastest growing techniques of Soft Computing, i.e. Artificial Neural Networks (ANNs) for obtaining various performance measures of a cellular radio system. A prioritized channel scheme with subrating is considered in which a fixed number of channels are reserved for handoff calls and in case of heavy traffic, these reserved channels are subrated into two channels of equal frequency to deal with more handoff calls. Two models dealing with infinite and finite number of subscribers are considered and the blocking probabilities of new and handoff calls are computed analytically as well as by using ANNs. A feedforward two-layer ANN is considered for obtaining the blocking probabilities. The backpropagation algorithm is used for training the ANN. The analytical and ANN results are compared by taking the numerical illustrations.

Keywords: Artificial Neural Networks, Cellular Radio System, Handoff, Reserved Channels, Subrating, Backpropagation

1. Introduction

Among the various paradigmatic changes in science and technology that have taken place in this century, one such change concerns the concept of Soft Computing (SC). Soft computing provides flexible information processing capabilities for handling real life ambiguous situations. Hard computing has the characteristics of precision and categoricity while the soft computing has the properties of approximation and dispositionality. Soft computing exploits the tolerance for imprecision and uncertainty to achieve tractability, lower cost, high Machine Intelligence Quotient (MIQ) and economy of communication.

One of the most powerful techniques of soft computing is Artificial Neural Networks, which aims to perceive and comprehend the significance of the data with which they are trained. ANN approach is frequently employed to analyze a variety of problems and is best distinguished from other SC techniques in that it is non-rule-based and can additionally be made stochastic so that the same action does not necessarily take place each time for the same input. A stochastic behavior allows a neural network to explore its environment more fully and potentially to arrive at a better solution than the conventional methods. ANN is a powerful data-modeling tool that is able to capture and represent complex input/output relationships. The properties of ANN like learning and adaptation, classification, function approximation etc. have made them of extreme use in solving various mathematical problems. Neural networks have been successfully applied to broad spectrum of data-intensive applications, such as Signal Processing [1], Chip Designing [2], optimization problems [3] and in many engineering problems [4]. ANNs are also used for solving problems that are too complex for conventional technologies e.g., problems that do not have an algorithmic solution or for which an algorithmic solution is too complex to be found.

There are multitudes of different types of ANNs. Some of the more popular include the Multilayer Perceptrons (MLPs), which are generally trained with the backpropagation algorithm [5]. This type of neural network consists of multiple layers and is known as a supervised network because it requires a desired output in order to learn. The goal of this type of network is to create a model that correctly
maps the input to the output using historical data so that the
model can then be used to produce the output when the
desired output is unknown. A three-layer feedforward ANN
with sigmoidal activation functions in the hidden layer and
trained using the backpropagation algorithm, is able to
approximate an arbitrary nonlinear function [6].

ANNs have been applied to the problem of traffic
prediction, adaptive control of nonlinear traffic etc. [7, 8, 9,
10, 11]. Researchers have used ANNs for bandwidth
allocation [12], admission control [13, 14] and for computing
the optimal number of channels to be allocated to various
users in GPRS [15]. Several researchers have also used ANN for location detection and prediction in cellular networks [21, 22].

In this paper, we consider a cellular radio system with a
prioritized scheme in which some channels are fixed exclusively
for the handoff calls. Also the reserved channels are subrated
into two channels of equal bandwidth for serving more handoff
calls in case of heavy traffic. A feedforward ANN with three
layers is employed to compute the blocking probabilities of new
and handoff calls. The backpropagation algorithm is used for
training the network. The rest of the paper is organized as
follows: In section 2, the basic architecture of an ANN is
described along with the backpropagation algorithm. The
analytical model for the cellular radio system is discussed in
section 3. In section 4, the ANN approach for computing the
blocking probabilities of the cellular system, is discussed. The
results obtained from the analytical method and ANN are
compared in section 5 by taking the numerical illustrations.
Finally, the conclusion is drawn in section 6.

2. Architecture of ANN

ANNs are closely modeled on biological processes for
information processing, including specifically the nervous
system, and the neuron. A mathematical model of the neuron is
depicted in Figure 1. It shows n inputs with associated weights \( v_j \)
\((j=1,2,\ldots,n)\) and the bias \( v_0 \). The output \( y \) can be expressed as

\[
y = \sigma\left( \sum_{j=1}^{n} v_j x_j + v_0 \right)
\]

where \( \sigma(.) \) is a differentiable function known as the
activation function which is selected differently in different
applications.

Figure 2 exemplifies a graphical representation of a three-
layer ANN. The first layer is known as the input layer with \( n \)
number of inputs and the second layer is known as the hidden
layer, with \( L \) number of hidden-layer neurons. The third layer
is known as the output layer with \( m \) number of neurons.

ANN with multiple layers are known as MLPs. The
computing power of MLPs is significantly enhanced over the
two-layer ANN which consists of only input and output
layers. The output of the three-layer ANN is given by

\[
y_i = \sigma_2\left( \sum_{l=1}^{L} w_{il} \sigma_1\left( \sum_{j=1}^{n} v_{lj} x_j + v_{l0} \right) + w_{i0} \right), \quad i = 1, 2, \ldots, m \tag{2}
\]

\[\text{Figure 1. Mathematical Model of a Neuron.}\]

\[\text{Figure 2. Three-Layer Artificial Neural Network.}\]
Step 1: Initialize $E = 1$ and $e = 0$;
Step 2: Present the input pattern $X$ to the ANN;
Step 3: Repeat Steps 4 to 8 until $E < \varepsilon$ or $e > NE$
Step 4: Initialize weights $v_i$ and $w_{ij}$ randomly;
Step 5: Compute the outputs of the two layers as

$$z_j = \sigma(\sum_{j=0}^{N} v_j X_j), \quad i = 1, 2, ..., L \text{ and } X_0 = 1;$$

$$y_i = \sigma(\sum_{j=0}^{m} w_{ij} z_j), \quad i = 1, 2, ..., m \text{ and } z_0 = 1;$$

Step 6: Compute the sum-squared error as

$$E = \frac{1}{2} \sum_{i=1}^{m} (Y_i - y_i)^2$$

Step 7: Update the weights in layers 2 and 1 respectively according to

$$w_{ij} = w_{ij} - \eta \frac{\partial E}{\partial w_{ij}}; \quad i = 1, 2, ..., m; l = 1, 2, ..., L;$$

$$v_{ij} = v_{ij} - \eta \frac{\partial E}{\partial v_{ij}}; \quad l = 1, 2, ..., L; j = 1, 2, ..., n;$$

Step 8: $e = e + 1$;
Output: Updated weights for the two layers.

3. Analytical Model for Prioritized Scheme in Cellular Radio System

We consider a cellular system with a prioritized channel scheme in which, a fixed number of channels are reserved exclusively for the hand-off calls. In order to deal with heavy traffic conditions, these reserved channels are also subrated i.e. a reserved channel is divided into two channels of equal frequency. Jain and Rakhee [20] studied cellular system with subrating. Two models are considered with finite and infinite subscribers respectively. Both models are discussed later in this section. The arrival rates of all the calls are assumed to be Poisson and the service times are distributed exponentially. The mean cell residence time and the call residence times also follow exponential distribution.

Following notations are used for mathematical formulation of the analytical model:

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>Number of subscribers</td>
</tr>
<tr>
<td>$C$</td>
<td>Total number of channels in the cellular system</td>
</tr>
<tr>
<td>$r$</td>
<td>Number of channels reserved for handoff calls</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Mean call-holding time</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Mean cell residence time of each portable</td>
</tr>
<tr>
<td>$\lambda_n$</td>
<td>Arrival rates of new calls</td>
</tr>
<tr>
<td>$\lambda_h$</td>
<td>Arrival rates of handoff calls</td>
</tr>
<tr>
<td>$\lambda_c$</td>
<td>Arrival rate of calls; $\lambda = \lambda_n + \lambda_h$</td>
</tr>
<tr>
<td>$P_0$</td>
<td>Steady state probability that there is no call in the system</td>
</tr>
<tr>
<td>$P_i$</td>
<td>Steady state probability that there are $i$ calls in the system</td>
</tr>
</tbody>
</table>

$B_n$ Blocking probability of new calls
$B_h$ Blocking probability of handoff calls

3.1. Model with Infinite Number of Subscribers (ISM)

In this model, the number of subscribers in the system is assumed to be finite. The steady state probabilities are obtained as follows:

$$P_i = \left\{ \begin{array}{ll} \frac{\lambda^i}{i!} P_0 & 0 \leq i \leq c - r \\ \frac{\lambda_c^{i-r}}{i!} \frac{\lambda_h^r}{\lambda_0^r} P_0 & c - r + 1 \leq i \leq c + r \end{array} \right. \quad (3)$$

where $P_0$ is computed by using normalization condition as

$$P_0 = \left[ \sum_{i=0}^{c-r} \frac{\lambda^i}{i!} + \sum_{i=c-r+1}^{c+r} \frac{\lambda_c^{i-r}}{i!} \right]^{-1} \quad (4)$$

3.2. Model with Finite Number of Subscribers (FSM)

In this model, the number of subscribers is taken as finite, i.e. $M$. The steady state probabilities are given by the following equations:

$$P_i = \left\{ \begin{array}{ll} \frac{M!}{i!} \frac{\lambda^i}{(\mu + \eta)^i} P_0 & 0 \leq i \leq c - r \\ \frac{M!}{(c-r)!} \frac{\lambda_c^{i-r}}{(\mu + \eta)^r} \frac{\lambda_h^r}{\lambda_0^r} P_0 & c - r + 1 \leq i \leq c + r \end{array} \right. \quad (5)$$

where $P_0$ is obtained as follows:

$$P_0 = \left[ \sum_{i=0}^{c-r} \frac{M!}{i!} \frac{\lambda^i}{(\mu + \eta)^i} + \sum_{i=c-r+1}^{c+r} \frac{M!}{(c-r)!} \frac{\lambda_c^{i-r}}{(\mu + \eta)^r} \right]^{-1} \quad (6)$$

Performance Measures

The blocking probabilities of new and handoff calls for both the models are calculated as

$$B_n = \sum_{i=c-r}^{\infty} P_i \quad (7)$$

and

$$B_h = P_{c+r} \quad (8)$$

4. The ANN Approach for Computing Blocking Probabilities

Now, we describe the ANN model for computing the performance measures of the cellular system discussed in the previous section. We consider a two-layer feedforward ANN with L neurons in the hidden layer and one neuron in the output layer. The activation functions at the hidden layer and output layer are assumed to be ‘sigmoid’ and ‘linear’
respectively. The backpropagation algorithm is employed for training the network. For studying the effect of different parameters on the performance measures of the analytical models, ANNs with different combinations of input and output neurons are used, which are described in figures 3-6. The ANNs described in figures 3a and 3b are used to study the effect of $\lambda_\nu$ on $B_n$ and $B_h$ respectively for both models where, $\lambda_\nu$ is the input neuron and $B_n$ and $B_h$ respectively are the output neurons. In the ANNs shown in figures 4a and 4b, $C$ is the input neuron and $B_n$ and $B_h$ are the outputs respectively. The ANNs in figures 5a and 5b have two input neurons i.e $C$ and $r$, and the output neurons are $B_n$ or $B_h$. For studying the effect of $M$ for FSM, the ANNs used have $M$ as the input neuron and $B_n$ and $B_h$ as the output neuron as demonstrated in figures 6a and 6b.
Figure 6b. ANN Model for calculating $B_h$ for FSM taking $M$ as input.

5. Numerical Experiment

In this section, we compare the analytical results obtained in section 3 with the ANN results by taking some numerical illustrations. Firstly, we determine the performance measures for the models ISM and FSM by using the analytical results. Then these results are validated by using the ANN models discussed in section 4. For illustration, we assume $C=30$, $r=2$ and the arrival rate of handoff calls to be 20% of that of the new calls, i.e. $\lambda_n=0.2\lambda_\nu$. For ISM, $\mu$ is taken as 0.015 and $\eta$ is assumed to be 0.006. For FSM, we take $\mu=0.15$, $\eta=0.6$ and $M=46$. For all ANN models, the learning rate (lr) is taken as 0.01. Other ANN parameters for various results are summarized in table 1. For all ANN models, the backpropagation algorithms are run on Pentium IV using MATLAB 5.2.

<table>
<thead>
<tr>
<th>Fig. No.</th>
<th>7(a)</th>
<th>8(a)</th>
<th>9(a)</th>
<th>10(a)</th>
<th>11(a)-11(d)</th>
<th>12(a) &amp; 12(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of Epochs</td>
<td>900</td>
<td>1000</td>
<td>894</td>
<td>800</td>
<td>1000</td>
<td>2000</td>
</tr>
<tr>
<td>No. of Epochs after which SSE is calculated</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>200</td>
<td>500</td>
</tr>
<tr>
<td>No. of neurons in hidden layer (L)</td>
<td>20</td>
<td>15</td>
<td>25</td>
<td>15</td>
<td>20</td>
<td>15</td>
</tr>
<tr>
<td>Error goal</td>
<td>$10^{-1}$</td>
<td>$10^{-2}$</td>
<td>$10^{-6}$</td>
<td>$10^{-3}$</td>
<td>$10^{-5}$</td>
<td>$10^{-11}$</td>
</tr>
</tbody>
</table>

Figures 7a and 8a exhibit the analytical as well as ANN results for $B_n$ and $B_h$ of ISM respectively by varying $\lambda_n$. Similarly, $B_n$ and $B_h$ for FSM by varying $\lambda_n$ are shown in figures 9a and 10a. Obviously, both $B_n$ and $B_h$ increase with $\lambda_n$ for both the models. The variation of the sum-squared error with the number of epochs for each computation is demonstrated in figures 7b-10b corresponding to the figures 7a-10a. We notice that SSE decreases with the increase in the number of epochs and finally SSE reaches the required error goal. The respective error surface graphs are also shown in the figures 7c-10c. These graphs represent those values of the weights and biases for the ANNs, which give the lowest error. In each of these graphs, we note that the error surface has a global minimum at the center of the plot and the side valleys lead to local minima.

Figures 11a – 11d display the ANN results for $B_n$ and $B_h$ for ISM and FSM by varying $C$ and $r$ both. We note that for both models ISM and FSM, $B_n$ decreases with $C$ and increases with $r$. Also, $B_h$ decreases with $r$ and is almost constant with $C$. These results are quite comparable with the analytical results.

Figures 12a and 12b depict the effect of $M$ on $B_n$ and $B_h$ respectively for FSM by taking analytical and ANN results as well. We notice the obvious result that both $B_n$ and $B_h$ increase with $M$ as expected.
Figure 8a. Bh by varying $\lambda n$ for ISM.

Figure 8b. SSE vs. Epochs for Fig. 8(a).

Figure 8c. Error Surface Graph for Fig. 8(a).

Figure 9a. Bn by varying $\lambda n$ for FSM.

Figure 9b. SSE vs. Epochs for Fig. 9(a).

Figure 9c. Error Surface Graph for Fig. 9(a).
Figure 10a. Bh by varying ln for FSM.

Figure 10b. SSE vs. Epochs for Fig. 10(a).

Figure 10c. Error Surface Graph for Fig. 10(a).

Figure 11a. Bn by varying C and r for ISM.

Figure 11b. Bh by varying C and r for ISM.

Fig. 11c. Bn by varying C and r for FSM.
cellular radio system. A prioritized channel scheme with subrating has been considered for the cellular system. The blocking probabilities of handoff and new calls have been determined by using a three-layer feedforward neural network. The backpropagation algorithm has been used for training the network.

The numerical simulations show that the results obtained by ANNs are comparable with the analytical results. We also conclude that once the ANN is trained against a data set, it takes less computational time than the conventional methods for calculating the required results which indicate that ANNs provide an easy and fast solution technique and are better than the conventional methods.

We have used ANNs for obtaining the performance measures of a cellular system. ANNs can be further used for taking handoff decisions for practical mobile cellular networks. Also, other soft computing techniques viz. Genetic Algorithms and Neuro Fuzzy Systems can be explored for modeling the performance of cellular networks.

6. Conclusion and Scope of Further Research

In this paper, we have investigated the potential of artificial neural networks for analyzing the performance of a

References


