Prediction of Corrosion Inhibitor Efficiency of Some Aromatic Hydrazides and Schiff Bases Compounds by Using Artificial Neural Network

Hanan A. Radhy Al.Hazam
College of Science, University of Basrah, Basrah, Iraq
Email: hananalhazam1965@yahoo.com

(NJC)
(Received on 25/5/2008) (Accepted for publication 2/3/2009)

Abstract
Artificial neural networks are used for evaluating the corrosion inhibitor efficiency of some aromatic hydrazides and schiff bases compounds. The nodes of neural network input layer represent the quantum parameters, total negative charge (TNC) on molecule, energy of highest occupied molecular orbital (E Homo), energy of lowest unoccupied molecular orbital (E Lomo), dipole moment(μ), total energy (TE), molecular volume(V), dipolar-polarizability factor(Π) and inhibitor concentration (C). The neural network output is the corrosion inhibitor efficiency (E) for the mentioned compounds.

The training and testing of the developed network are based on a database of 31 published experimental tests obtained by weight loss. The neural network predictions for corrosion inhibitor efficiency are more reliable than prediction using other conventional theoretical methods such as AM1, PM3, Mindo, and Mindo-3. Key word: Neural network, Corrosion inhibitor efficiency.

الخلاصة
استخدمت تقنية الشبكة العصبية الصناعية في تقييم كفاءة تثبيط التآكل لبعض مركبات الهيدرازايد الأروماتية وقواعد شف. تضمنت طريقة الاستقبال الشبكة معاملات كمية هي الشحنة السالبة الكلية (NTC) ،طاقة أعلى اوربيتال مملوء (E Homo) ،طاقة اوربيتال فارغ (E Lomo) ،عمق ثنائي القطب (μ) ،الطاقة الكلية (TE) ،الحجم الجزيئي (V) ،عامل الاستقطاب (Π) وتركيز التآكل (C) للمركبات المذكورة أعلاه. واعتمد في تدريب واختبار الشبكة 31 قيمة عملية ناتجة من فقدان الوزن. ووجد من النتائج المستحيلة تثبيط التآكل ان تقنية الشبكة العصبية أكثر دقة من المحاسبة بالطرق النظرية الأخرى مثل AM1،PM3،Mindo3،Mindo،PM3،AM3.
Introduction

Some o-, m-, and p- substituted benzaohyrazides, cinnamohyrazide, and Schiff bases derived from substituted benzaldehydes with 2-aminopyridine and 2-aminopyrimidines, benzaldehyde and cinnamaldehyde with p-phenylenediamine were used as corrosion inhibitor of steel in the presence of HCl medium [1].

The efficiency of an organic inhibitor of metallic corrosion does not only depend on the structural characteristics of the inhibitor but also on the nature of the metal and environment. The selection of suitable inhibitor for a particular system is a difficult task because of the selectivity of the inhibitors and a wide variety of environments.

Elashry et.al.[1] have found relationship between the inhibitor efficiency for hydrazides and Schiff bases with quantum chemical calculation parameters including total negative charge (TNC) on molecule, energy of highest occupied molecular orbital (E Homo), energy of lowest unoccupied molecular orbital (E Lomo), dipole moment (μ), total energy (TE), molecular volume(V), dipolar-polarizability factor (Π) and inhibitor concentration (C). The correlation analysis between quantum parameters mentioned above and experimental corrosion inhibitor efficiencies (E exp) based on the weight loss method shows significant correlation ($\rho < 0.001$).

The inhibitor efficiency increases if the compound can donate electrons from its Homo to Lumo of the metal whereby chelating on the metal surface occurs [1, 2].

Artificial neural networks (ANN) are an artificial intelligence tools that were proposed for modeling systems that have complex nonlinear input/output relationships. Neuron computing, a technology of ANN, is a powerful tool for solving nonlinear problems that involve mapping input data to nonlinear output data without having any prior knowledge about the mathematical process involved. ANN has been employed successfully in different applications including engineering, medical, insurance process and scientific environment [3-5].

In this paper, an attempt is made to use ANN to predict the corrosion inhibitor efficiency (E) of any compound provided that input data are within the range used in the training set.

Construction of ANN Model

It was recommended [3] that a multilayer feedforward backpropagation neural network is the most widely used network for its efficient generalization capabilities. A typical multilayer feedforward network is presented in Figure (1). This type of neural network consists of an input layer, one or more hidden layer (s), and an output layer. Layers are fully connected, as shown in Figure (1), by arrows, and comprise a number of processing unit, the so–called nodes or neurons. The strength of connection between neurons is represented by numerical values called weights (W). The optimum number of hidden layers and the number of neurons in each layer is problem specific. Therefore, trial and error should be carried out to choose an adequate number of hidden layers and the number of neurons in each hidden layer [6]. The smallest network unit (the neuron) receives its input through a connection that multiplies the value of input by the weight (W) and adds a bias (b). The sum of weighted inputs and their weights and biases is the argument for a transfer function (f) that produces the neuron output. The pattern of connectivity in the network is represented by weight vector W. The
initial values for weights and biases of
the network can be arbitrary chosen.
By adjusting the weights and biases,
the network can exhibit any desired
output. The process of adjusting the
weights and biases is known as
training. In other words, an ANN
learns from examples (of known
input/output sequences) and exhibits
some capability for generalization
beyond the training [7].

Transfer functions for the
neurons are needed to introduce non-
linearity into the network. Transfer
functions commonly used in
feedforward neural networks include
linear, log-sigmoid, and tan-sigmoid
transfer functions [7]. These transfer
functions have outputs ranging
between 0 and 1 suitable for
backpropagation networks because they
are differentiable [7].

Training Process and Topology of the Neural Network

The ANN was designed using the
Levenberg–Marquardt algorithm, built
in MATLAB version 7 [8]. For
backpropagation algorithm in
MATLAB version 7 the data set was
dividing into two sets; training set, and
testing set. The training set is used to
gradually reduce the ANN error. The
error is monitored during the training
process. The training set error will
normally decrease during training. The
test set is used as a further check for
the generalization of the network
(NN), but do not have any effect on the
training. In the present study, training
data set comprises 23 data entries and
the remaining data entries 8 are used as
testing sets. The dividing process was
carried out randomly between the two
sets and each dataset has been
examined to ensure that it converse the
range of input parameters.

Several network models were
created by varying the number of
hidden layers, number of neurons in a
hidden layer, and training parameters
of Levenberg–Maquardt algorithm. After a number of trials, the best
neural network was determined to have
four layers: input layer of 8 neurons,
two hidden layers, the first of 5
neurons, the second of 8 neurons, and
an output layer of 1 neuron. The 8
input neuron represent: total negative
charge (TNC) on molecule, energy of
highest occupied molecular orbital
(E Homo), energy of lowest
unoccupied molecular orbital (E
Lomo), dipole moment (μ), total
energy (TE), molecular volume (V),
dipolar-polarizability factor (Π) and
inhibitor concentration (C). The
output neuron gives the corrosion
inhibitor efficiency (E). Table (1)
shows the ranges of each different
parameter in the study. All calculations
of the input and output are obtained
from Ref. [1].

Performance of ANN

The performance of the
developed neural network was
monitored during the training process
as the mean absolute error over all the
training data. Error was estimated for
each point as difference between
efficiency (E) output and experimental
(E exp). The mean squared error
function was used as the performance
function of training the current neural
network. The training process stops
when any of the following conditions
are satisfied: the maximum number of
iteration (epochs) is reached; the
performance has been minimized to
the required target; the performance
gradient falls below a minimum value.

Normalizing Input and Output Data Set

Normalization of input and output
data sets within a uniform range before
they are applied to neural network are
essential to prevent large number from
overriding smaller ones, and to prevent
parameter saturation of hidden nodes,
which impedes the learning process. The limitation of input and output
values within a specified range are due to large difference in the values of data provided to the network. Besides, the activation function used in the backpropagation neural network is hyperbolic tangent function, the lower and upper limits of this function are -1 and +1 respectively. The used function for normalization is

\[ Z_i = 2[(X_i - X_{\text{min}}) / (X_{\text{max}} - X_{\text{min}})] - 1 \]

where \( X_i \) is the value of any variable

\( X_{\text{min}} \) = the minimum value of \( X_i \)

\( X_{\text{max}} \) = the maximum value of \( X_i \)

**Results and Discussion**

The neural network is trained with different learning rate values (\( \eta \)) and momentum coefficient (\( \mu \)) which are important parameters that control the effectiveness of training algorithm using the steepest descent algorithm (Gradient Descent, GD) with momentum. The network performance can be improved by finding optimal value for \( \eta \) and \( \mu \). The effective values for both the learning rate and momentum coefficient are 0.5 and 0.9 respectively. These values give the least mean square error (MSE).

The performance for training and generalization (test) sets are simulated using GD, as shown in Figure 2. The network was trained for 2000 epochs to check if the performance (MSE) for either training or testing sets might diverge\(^9\).

Regression analysis between the output of the neural network and the corresponding target has been performed using the routine "postreg" in Matlab ver. 7. The format of this routine is \([m,b,r] = \text{postreg}(a,t)\), where \( m \), \( b \) correspond to the slope and the intercept of the best linear regression that relates the targets to the network outputs. If the fit is perfect (outputs exactly equal to targets) the slope would be 1 and the intercept with Y-axis would be 0. The third variable, \( r \), is the correlation coefficient between the outputs and targets. It is a measure of how well the variation in output is explained by the target. If the number equal to 1, then there is perfect correlation between the targets and outputs. Fig. 3 shows that the correlation coefficient is equal to 0.968; this indicates that the neural network approach is a reliable method and can give very accurate results.

The study has demonstrated that the neural network can effectively generalize correct responses that only broadly resemble the data in the training set. The neural network can now be put to use with the actual data, this involves feeding the neural network the values for TNC, E Homo, E Lumo, \( \mu \), TE, V, \( \Pi \) and C. The neural network will produce almost instantaneous results of corrosion inhibitor efficiency (E). The predictions should be reliable, provided the input values are within the range used in the training set.

**Table 1 Ranges of parameters in database**

| Parameter                          | Ranges                      |
|------------------------------------|-----------------------------|
| Total energy (TE)                  | [-1739.6 - (-2089.8)] ev    |
| Inhibitor concentration (C)        | [0.2 – 1 ] mM               |
| Molecular volume (V)               | [77 - 76.3]cm\(^3\) /mole  |
| Energy of Homo orbital (E Homo)    | [-7.85 – (-10.1)] ev        |
| Energy of Lomo orbital (E Lomo)    | [-0.07 – (-0.992)] ev       |
| Dipole moment (\( \mu \))         | [1.315 -6.27] Debye         |
| Dipolar-polarizability factor (\( \Pi \)) | [1.43-2.89]*              |
| Total negative charge (TNC)        | [-1.342 – (-2.812)] ev      |

*calculated by using Hicky and Possino- Reader [9]
Fig. (1) Structure of artificial neural network

Fig. 2 Convergence of the NN for training and testing sets based on the GD with momentum – algorithm
Fig. 3 Comparison between NN result and target result for training patterns based on GD algorithm

Conclusions

This study demonstrated the feasibility of using a simple back-propagation neural network to model the corrosion inhibitor efficiency for aromatic hydrazides and Schiff bases compounds. After learning from a set of selected patterns, the neural network models were able to produce reasonably predictions. Actual field data were modeled using networks. The neural network model was found to be more reliable than other conventional methods.

References

1- Elsayed H. E.; Ahmed E. N.; Samy A.N.; and Salam R. J., *Arkivie*, 2006, 26, 220.
2- Quraish M. A.; Sardar R. and Jamal D., *Mater Chem. Phys.*, 2001, 71, 309.
3- Edward J. Rzempoluck, Neural Network Data Analysis Using Simulnet, Springer, New York, 1998.
4- Antony T.C., *Aci Structural Journal*, 2002, 28, 92.
5- Kuchman A.D., Artificial neural network for image understanding, Van Nostraud Reinht, NY, USA, 1994.
6- Mathworks Inc "Matlab the Language of Technical Computing" Ver.7 Natick, 1999.
7- Hyykin S., Neural network: A Comprehensive Foundation, Prentice Hall Inc. , Englewood Cliffs. N. J. 1999.
8- Jang j.; Sun C. and Mizutani E., Nearo Fuzzy and Soft Computing ,Computational approach to learning and machine intelligence prentice hall- Inc., Englewood Cliffs 1997.
9- Yurt A.; Bereket G. and Kiroad A., *J. Appl. Electrochem.*, 2005, 35, 1025.