Pattern recognition and data mining software based on artificial neural networks applied to proton transfer in aqueous environments

Amani Tahat[a)†], Jordi Marti[a)], Ali Khwaldeh[b)], and Kaher Tahat[c)]

[a)] *Department of Physics and Nuclear Engineering, Technical University of Catalonia–Barcelona Tech, B4-B5, North Campus UPC, 08034 Barcelona, Catalonia, Spain*

[b)] *Department of Computer Engineering, Faculty of Engineering, Philadelphia University, Amman, Jordan*

[c)] *Department of Graduate Business Studies, Waterford Institute of Technology, Main Campus Cork Road, Waterford, Ireland*

(Received 23 June 2013; revised manuscript received 3 October 2013; published online 10 February 2014)

In computational physics proton transfer phenomena could be viewed as pattern classification problems based on a set of input features allowing classification of the proton motion into two categories: transfer 'occurred' and transfer 'not occurred'. The goal of this paper is to evaluate the use of artificial neural networks in the classification of proton transfer events, based on the feed-forward back propagation neural network, used as a classifier to distinguish between the two transfer cases. In this paper, we use a new developed data mining and pattern recognition tool for automating, controlling, and drawing charts of the output data of an Empirical Valence Bond existing code. The study analyzes the need for pattern recognition in aqueous proton transfer processes and how the learning approach in error back propagation (multilayer perceptron algorithms) could be satisfactorily employed in the present case. We present a tool for pattern recognition and validate the code including a real physical case study. The results of applying the artificial neural networks methodology to crowd patterns based upon selected physical properties (e.g., temperature, density) show the abilities of the network to learn proton transfer patterns corresponding to properties of the aqueous environments, which is in turn proved to be fully compatible with previous proton transfer studies.

Keywords: pattern recognition, proton transfer, chart pattern, data mining, artificial neural network, empirical valence bond

PACS: 61.20.Qg, 31.15.xv, 07.05.Mh **DOI:** 10.1088/1674-1056/23/4/046101

1. Introduction

In general, charts can be used to formulate physical information (i.e., quantum information) that is contained in a physical system and can even be the only tool that a researcher utilizes. Even though creating charts and finding patterns on charts were difficult before the advent of computers and data feeds, with the advancement of technology and the increased popularity of data mining technical analysis, the use of charts has greatly increased, making them one of, if not the most important, tools used by scientific analysts to study physical phenomenon, such as the proton transfer phenomena in aqueous environments, which will be considered here, and the concept of the quantum entanglement for the effectively describing of direct or causal relationships between apparently distinct or spatially separated particles. Hence, chart patterns refer to the graphical representations of the output data of specific software, which is being used in the required simulation. Achieving the best results of using charts requires correctly identifying chart patterns on the required chart, for distinguishing one thing from another; so researcher must be able to identify chart patterns properly by defining methods of recognition using new technological pattern recognition tools. An important point before we proceed, patterns itself could be defined as regularities in the output data when using computer simulation or an experimental results; however, the problem of searching for patterns in a large set of data is an essential problem of science and has a long and successful history.[1] For example, in the sixteenth century the empirical laws of planetary motion were discovered by Johannes Kepler who had examined data on the motion of the planets for finding patterns known as (Kepler's laws) based on the general astronomical observations of Tycho Brahe, which in turn provided a target for the development of the classical mechanics. These patterns had been used also by Newton for developing his law of gravity and providing an explanation for the patterns uncovered by Kepler. Therefore, pattern recognition could be recognized as a key part of scientific investigation. One more example can be shown in developing the periodic table of Dmitri Mendeleiev, who found patterns in the properties of the elements. Similarly, regularities in atomic spectra have been discovered in the early twentieth century and later on played a key role in the development of the quantum theory of the atom and so forth.[2]

1.1. Automated pattern recognition

The automatic discovery of regularities in data in the field of pattern recognition has become relevant through the use of computer algorithms and the analysis of these regularities by classifying the data into different categories; therefore, this

†Corresponding author. E-mail: amanitahat@yahoo.com

© 2014 Chinese Physical Society and IOP Publishing Ltd

problem has been considered and studied in many situations (e.g., medicine, finance, robotics, physical phenomena). When the automatic identification of patterns is required, the aim of pattern recognition is the same in all of these situations and the pattern recognition would focus on classifying data patterns according to either the extracted statistical information from the patterns or some of priori knowledge for providing a reasonable answer for all possible inputs and doing the "fuzzy" matching of inputs. Most commonly, the patterns to be classified depend on using groups of observations and measurements, which serve as defining points in a relevant multidimensional space. For example, imagine a two-dimensional data set of the heights and weights of some different animals if there are two types of animals in that data collection as presented in Fig. 1(a). The green circles show the examples of animals of class 1 (snarks) and the yellow diamonds show examples of animals of class 2 (boojums). Given a new data point, which type of animal is it? In this case, someone would look at some methods that draw a straight line between classes – linear classifiers and then the decision boundary (dashed line) would present the set of points at which both classes would be equally likely.

Nevertheless the problem might be nonlinear as illustrated in Fig. 1(b). In such a case we would need different classifying methods, for example the implementation of the approach to learning in a multilayer perceptrons (MLP) network which is the most suitable type of the artificial neural networks (ANNs).[3] ANNs mimic human brain when coping with incomplete and confusing information sets. Another important issue with implementing the ANN method here is that it is based on supervised learning which requires a data set, in many situations of considerable size, for which the target output is known taking into account the correlation between the inputs.[3] Output refers to an experimental data or output of computer simulation that works as an input of ANN network to start to analyze and classify the data, which means that ANN allows learning by example. Therefore, to find an automated method for classifying data, we need a way of plotting data; so that the classes would be separated as shown in Figs. 1(a) and 1(b) respectively, then the need of a visualization tool would be important to see the characteristics of the data after finding ways of incorporating background, a priori knowledge about the problem, into a specific method. Table 1 shows some general examples of pattern recognition applications.

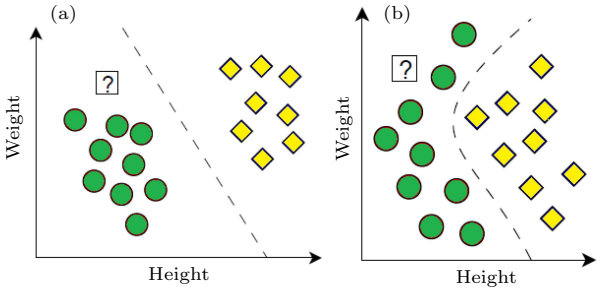


Fig. 1. (color online) Different patterns in data analysis.

Table 1. Examples of pattern recognition practical applications.

Problem domain	Application	Input pattern	Pattern classes
Data mining	Searching for meaningful patterns	Points in multidimensional space	Compact and well-separated clusters
Biometric recognition	Personal identification	Face, iris, fingerprint	Authorized users for control of access
Document image analysis	Optical character recognition	Document image	Characters, words
Document classification	Internet search	Text document	Semantic categories
Document classification	Spam mail filtering	Email	Spam/non-Spam
Speech recognition	Telephone directory assistance	Speech waveform	Spoken words
Medical	Computer aided diagnosis	Microscopic image	Cancerous/healthy cell
Industrial automation	Fruit sorting	Images taken on a conveyor belt	Grade of quality
Remote sensing	Forecasting crop yield	Multispectral image	Land use categories
Bioinformatics	Sequence analysis	DNA sequence	Known types of genes

1.2. Artificial neural networks

In terms of the type of ANN applications, implementing ANN represents the technique of data analysis probably due to an increased complexity of the data (e.g., chart patterns), in addition to physics process investigated which imposes more demands on the analysis techniques. Although many applications are possible, there is a little work that can be found in the literature. The first application from which a physics result was extracted with using ANN was for the decay of the Z boson,[4] and the results were used further for determining the decay probability of Z into the corresponding states. A good review of these applications until 1999 was published in Ref. [5]. Other important physics results were obtained with such ANN applications in high-energy physics.[6,7] In the meantime there were so many several data analysis applications that used standard method instead of using ANNs for automatic estimation of data and chart patterns in physics and some other relevant fields such as those mentioned in Refs. [8]–[12]. In all applications ANNs gave better results than standard methods, which is mainly due to the highly non-

linear characteristic of the method.[3] In this regard, there are many new opportunities for using ANNs in physics nowadays, mapping sets of physical problems to analogous applications in other areas of science as well as engineering, such as medical diagnosis disease applications,[13] where ANNs show significant results in dealing with data represented in symptoms and images, i.e. chart patterns. On the other hand, this study will provide an interesting and useful ANN application for predicting proton transfer dynamics in aqueous environments.

1.3. Proton transfer in aqueous environments

This is a fundamental process in many biological and chemical processes with interesting technological applications, from energy production to aerosol industry and to atmospheric chemistry. It is a very useful phenomenon in all biomedicine researches in different fields, such as breathe and urine analysis for disease monitoring and diagnosis for diabetes as well as heart diseases. Among others, some important cases of proton transfer (PT) in aqueous environments are as follows.

(i) Proton transfer in water between different water molecules, namely intermolecular proton transfer, could occur in a wide variety of thermodynamically conditions. It was assumed to be due to a "Grötthus mechanism", which consisted in a proton hopping between neighbor water molecules linked by hydrogen bonds.[14] A similar process occurred in acids.

(ii) Intermolecular proton transfer, occurring in a variety of cases, such as guanine,[15] curcumin in reverse micelles[16] or in aromatic carbons.[17]

(iii) Chemistry of excited states in organic compounds and in drugs, process of central importance in biochemistry and molecular biology.[18]

(iv) Recently, researchers at the Lawrence Berkeley National Laboratory found that proton transfer in DNA might happen even when hydrogen bonds were blocked, which would lead to a new class of proton transfer.[19]

Upon the above explanations, using chart patterns to support theoretical investigations of proton transfer (PT) in aqueous environments via high performance computational codes is of great interest to understand the adaptive phenomenon. Some standard tools for such a simulation are acid-base packages,[20] Car-Parrinello simulations[21] or Multi-State Empirical Valence Bond (EVB) codes.[22,23] Among others the present study adapted the development of an EVB code as a powerful code, which had been proved to be a very successful tool for the simulation of the proton transfer in condensed media and it could be used for a variety of solvents, from pure water to mixtures in order to model proton transfer processes for water complexes, as well as for acids. In this regard, the EVB method has been developed first by Warshel and Weiss in 1980[24] who have recently been awarded the 2013 Nobel prize in chemistry for developing it by using both quantum and classical physics to develop computer models of chemical systems that efficiently reveal the details of chemical reactions and other molecular activity on a nano scale. The EVB method could be considered as one of the first top, distinctive and unprecedented, computer simulation techniques in this field which has been described extensively in Ref. [24]. Accordingly, using the EVB code for such PT simulations is mainly based on analyzing the chart patterns of its output data, even though there are some obstacles that reduces its efficiency, as it does not support plotting charts and it has a very huge set of tabulated output data that need to be extracted from and exported manually to any plotting program (e.g., MS-Excel). This consumed more time and required greater attention for different quality characteristics, though, for large files it became tedious and error prone because plotting programs sometimes could not cope with very large files directly and had some limitations. For example, MS-Excel could not draw charts for large input files directly where the maximum number of data points that could be used in an excel data series for a two-dimensional (2D) chart must not exceed 32000.[25] Moreover plotting a large set of EVB output data will give noisy and crowd charts, and finding patterns would be impossible (e.g., up to 10^8 records in a table). For the sake of concision, this study will be restricted to a description on how to design a learning machine tool, which mapped sets of input data onto a set of appropriate output for recognizing EVB chart patterns automatically based on ANNs networks.

2. Patterns on charts using original EVB output data

Consider the pure water as an example of aqueous environments to show a sample of such EVB chart patterns in a physical system. In this example the EVB code will provide a long simulation for proton transfer when being placed in bulk water that consists of 125 water molecules to study the microscopic behaviors of water by means of a series of molecular dynamics simulation. In aqueous proton transfer, two dominant structures arise, namely the Eigen ($H_9O_4^+$)[26] and Zundel ($H_5O_2^+$)[27] solvation structure patterns, both complexes being the result of the solvation of the primary hydronium water complex (H_3O^+). In both cases, PT could be viewed as a pattern classification problem based on a set of EVB input features (temperature, density, number of water molecules) to classify a proton transfer as 'transfer occurred' or as 'not occurred'.[28] Moreover the output chart will show two different chart patterns during this long simulation: a pattern that indicates proton transferred and a pattern that indicates proton non-transferred as a function of simulation time under a specified temperature. Such chart patterns are shown in Fig. 2.

During the simulation as we could read from Fig. 2, each pivot will have an index, to determine the instantaneous tagged water molecule. Flat patterns indicate no transfer, whereas, permanent change of pivot label indicates that a PT has occurred. Spikes represent aborted transitions. Once characterizing the solvation of hydronium ion in water and the corresponding proton transfer episodes, chart patterns that represent 'transfer occurred' are the required information needed for technical analysis, in order to extract the chemical properties of this system. Herein, the excess proton will attach itself to one of 125 water molecules, forming a hydronium cation (H_3O^+) with dynamical label along the simulation; many separate pieces of information may share the same pattern, and give the same shape for a series of proton transfer that may attach itself to the remaining water molecules. For example, as in this simulation the proton is attached exclusively to a little group of molecules tagged with 46, 49, 100, and 20. When coming to analyze such chart patterns, we can say that all patterns are copies of each other and do not give a clear pattern information when trying to determine the tagged water molecule, which carries the excess proton in the system as a function of simulation time. In other words, such EVB output graphical representations could not make predictions of proton transfer movement (transferred, non-transferred). Extracting the information beyond such chart pattern is complicated because it is a crude chart pattern, and presents a totally unclear (unpredictable) scenario of proton movement along with hydrogen-bonded water molecules, where the motion could be described with the wired line between two flat peaks. It is essentially a statistical problem, with the particularity that the underlying mechanism of PT is a quantum process. So, a correct study will need to produce more charts by changing the temperature and again that would give another chart pattern, in general many simulations will give patterns that seem to be similar in nature, but in fact no two patterns are exactly alike. The challenge here is how to identify the specific pattern, which helps to determine the molecular chemical properties, such as proton transfer rates, diffusion coefficients of the proton, water local densities at different temperatures, and so on. Finding a normal fitting linear or logarithmic line for such a chart is impossible by using automatic fitting techniques of chart plotting software (e.g., origin, excel). Therefore, it is difficult to find pattern on such charts to enable extracting information (e.g., slope), then a pattern recognition technique would be needed here for plotting and controlling the required smooth and clear charts that leads to better visualizing and analyzing. Coming back to the chart pattern of Fig. 1(b), and comparing it with chart patterns in Fig. 2, we find that they illustrated the same case at the computational level. So, we can assume that the ANN is an efficient approach to be used to solve this problem that does not have algorithmic solutions in which inputs and outputs variables do not have a clear relationship between them (e.g., temperature and *pivot* has a non linear relationship). Therefore using ANNs for handling these proton transfer nonlinear functions in the data filtering, data mining as well as chart pattern recognition to find a solution is adopted here.

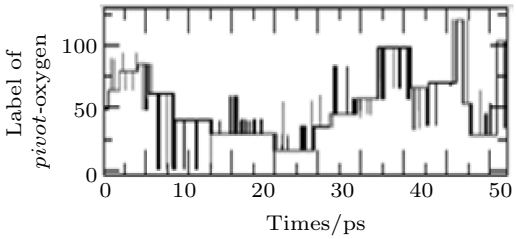


Fig. 2. Graphical representation of original EVB code output data representing the time evolution of the hydronium (H_3O^+) molecule label (called *pivot* from here) in pure water at room temperature.

3. Physical mechanism of the method

We propose a chart pattern recognition tool that is optimally programmed by using Python to work as a first helpful ready-to-use add-on tool of the EVB code. This tool is expected to provide a better visualization of the graphical representation of EVB output data, plus enabling automatic recognition of patterns on charts based on well-known filtering, correlating, and classification functions. It is designed to serve two purposes: data mining and chart pattern recognition, to be used with independent goals at each stage of the process to control charts in monitoring and filtering EVB output files with more accuracy, therefore, if any 'out of control' condition should be found, the reasons for the variation outside the stable patterns would be discovered. Afterwards, the proposed tool will simply enable the technical analysis of several important chemical properties, by means of the EVB code output graphical data, having the aim of enhancing the performance and efficiency for systems of a large number of molecules and for atomic open source codes (e.g. EVB code; AJAC code[29]) to be published freely with its documentations including the binary file of the EVB code for public benefit under the GNU GPL license, so that people can redistribute it, improve it, and describe it. It will be designed to work under multi-operating systems including UNIX, LINUX, WINDOWS, MAC, and Android. In this pattern recognition tool, ANN with back propagation is applied to the literature data generated by EVB code, and stored in a database to construct a classifier system which could be used to predict three chemical properties of proton transfer phenomena in aqueous environments based on chart patterns. These properties are the proton transfer rate μ, the hydronium–water local density field ρ, and the diffusion coefficient of proton, D. In other words, this model would be a physically basic model. Firstly, EVB must be run to generate data of possible conditions based on changing the input param-

eters to compose the output data sets in the database. Subsequently, these data would be used as input in the current ANN model. Of course, the model techniques could only pickup the required behavior if the available physical condition is accurate and describes the relevant process in detail. The challenge of composing the input file of the tool by choosing the right data and the right examples for the training phase is often the most complicated part of using the ANN pattern recognition technique. The data must be processed before using the ANN approach, and there are some modifications that must be made to the data set to make it perform better in the training phase. Herein, from the normalized database, user needs to train the network where the ANN will be trained with N data points and tested (or validated) with N data points based on the case study and the selected chemical properties to be predicted, for example, how to predict the proton transfer rate based on the chart patterns, as it would be fully explained in the section of case study as presented in Fig. 3.

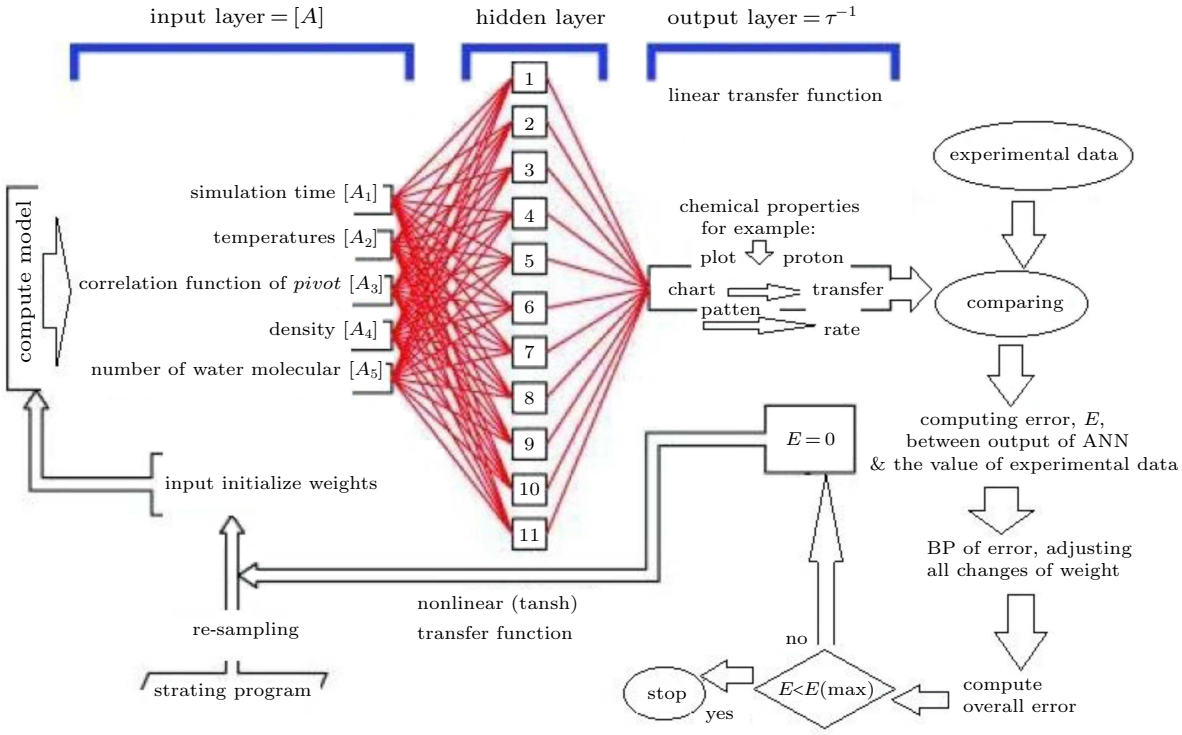


Fig. 3. (color online) Optimal ANN structure, together with a flowchart of the back propagation (BP) algorithm for the prediction of proton transfer rate, plus the basic algorithm loop of the proposed tool, which will be fully explained in the case study.

3.1. Source of the data set

As the tool will provide an easy-to-use database, where the original EVB collected data would be represented in the form of database files (i.e., output of EVB code) after several simulations as shown in Fig. 4, these data might be then cleaned or reduced by using ANN data mining techniques; feature extraction algorithms and the feature space dimension might be reduced; the system would then use description algorithms or classification algorithms based on the requested diagnostic purpose.[30] For instance, the system could predict the proton transfer and describe patterns in different classes depending on their characteristics thus to cope with the nonlinearity of the EVB input and output (see Fig. 3).

3.2. Computing process

Generally, the computing processes of the ANN algorithm are typically defined by three types of parameters: the interconnection pattern between different layers of neurons, the learning process for updating the weights of the interconnections, and the activation function that converts a neuron weighted input into its output activation. Consequently, in the computational process, the proposed Python tool would allow the operating BP algorithm in two phases via its graphical user interface (GUI):

1) Training phase, where the training data samples would be provided at the input layer for training the network with predefined set of data classes. The ANNs would be trained using a BP algorithm, based on the concept of improving the performance of network by reducing the error from the output data.[31] Once the network is trained, it could be saved for further use. The program uses data files, which have been computed using EVB, code as the main source of data for pattern recognition of proton transfer and it could be considered then as time series data of the selected required input. This is true for all other input and output parameters (e.g. number of chemical species, mass, number of atoms of each species, temperature, simulation time, density correlation function of

pivot, etc). In short for training MLP algorthim the problem of estimating the weights consists of the following parts: initializing the weights randomly; computing the derivative of the error function with respect to the weights via the error back propagation algorithm; updating the estimated weights via the gradient descent or scaled conjugate gradient method.[32,33]

2) Testing phase, during this phase the input layer would be provided with random test data (experimental) to predict the applied patterns. Several experimental data are applied to the new tool database, and the ANN results are compared with the experimental values in the testing phase for providing validity and accuracy.[32]

For more explanation, the user of the proposed tool will start to provide the algorithm with examples of target output data; furthermore the network will change the network weights so that when training is finished, the user will obtain the required output for a particular input. Each input pattern of this training set would be applied to the input units and then it would be propagated forward. The pattern of the activation arriving at the output layer would be compared with the correct (associated) output pattern for calculating an error signal. The error signal for each such a target output pattern would then be back propagated from the outputs to the inputs in order to appropriately adjust the weights in each layer of the network. The procedure would be repeated until the error function would reach the range of the error tolerance factor that has been set by the user. After back propagation network is learned, the correct classification for a set of inputs could be tested on a second set of inputs to see how well it classifies untrained patterns. In this case, the back propagation network could make a reasonable prediction about what data are missing from the current available information of a hidden data, which is a very difficult task to do in the real world, but the use of BP technique of this tool would allow user to guide the tool to find some corrected pre-information (such as desired output) as shown in Fig. 4.

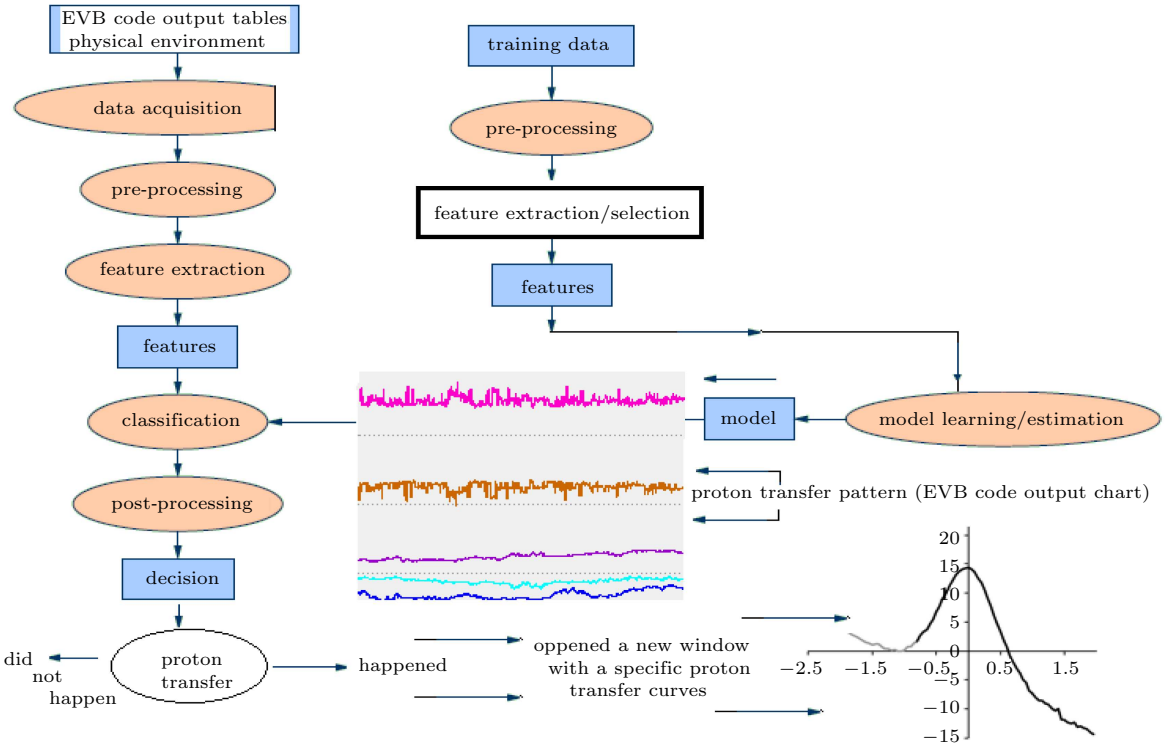


Fig. 4. (color online) Object/process diagram of the proposed chart pattern recognition system following the typical pattern recognition system architecture of Schalkoff.[34]

3.3. General steps of the data set preparations

This part is very important because the ANN pattern recognition neural tool is only as good as the quality of the data set; in this part user can create the ANN input file by resampling output files of EVB code available in the database. The proposed tool will consider a number of factors as input, in order to produce an output, which will give the result of whether a proton has suffered a transfer episode. In many computational methods any small change in the input pattern might impose a drastic change on the desired output pattern,[35] so that in order to obtain the best classification performance evaluations, this tool provides two kinds of data modification techniques, which could be easily controlled from the main window on click. They are the ANN modification parameters of learning and testing and the ANN input modification functions.

3.3.1. ANN modification parameters of learning and testing

As the accuracy of the ANN network output is dependent on the accuracy of the analytical numerical model used for training this network,[36] user of this tool can continue examining the behavior of changing the value of modification parameters of learning and testing algorithms on the output results by comparing with an available experimental value, thus investigating under what conditions the pattern recognition tool could effectively be used for predicting the proton transfer's chemical properties. Accordingly, these fitting parameters would be some constants for adjusting the algorithm of computational process[35,36] from the main window related to

i) The construction of the neural network itself (e.g. the number of hidden layers, the number of neurons in each hidden layer, the nature of the node nonlinearity that is used in the hidden layer (sigmoid or tansh));

ii) The generalized delta rule (e.g. learning rate, momentum term, and the set of initial weights).

3.3.2. ANN input modification functions

These functions could be defined as regularities in the output data of EVB code to create the required chart patterns following statistical functions. In other words these functions are a technique to modify the ANN input data set. Firstly, the tool will extract data set from the large EVB output tables subsequent to any specified case study, then the user will determine the relevant fitting function on click from the main window to generate the required fitting input to be learned and tested via neural network, any time user could adjust the two fitting techniques spontaneously, and continue attempting until reaching a smooth chart pattern instead of the original crowd chart pattern of EVB data that are shown in Fig. 2. On the other hand the tool provides three types of fitting functions (physical regularities on PT) as follows.

3.3.2.1. Population relaxations of *pivot* and proton transfer rate

This tool will provide a good technique to enable plotting smooth charts patterns to study the nature of proton transfer dynamics in different aqueous environments. Therefore, figure 2 could be significantly improved by performing an analysis based on time correlation functions following the literature study,[37] and by focusing on temporal correlation of type

$$C(t) = \frac{\langle \delta h_i(t) \cdot \delta h_i(0) \rangle}{\langle (\delta h_i)^2 \rangle}, \tag{1}$$

where $h_i(t) = 1$, if the hydronium label 'NH_3O' at time t corresponds to the i-th water molecule and $h_i(t) = 0$ otherwise. In the above equation, $\delta h(t) = h(t) - \langle h \rangle$ is the instantaneous fluctuation of observable $h(t)$ at time t away from its equilibrium value. The power of chart pattern recognition will appear in the last smooth chart so that plotting this chart with respect to time will represent a fitting chart and smooth chart, thus it will simply allow calculating the slope as

$$k_{\mathrm{PT}} = \lim_{t\to\infty}\Big(-\frac{\mathrm{d}\ln C(t)}{\mathrm{d}t}\Big). \tag{2}$$

According to Onsager's regression hypothesis,[38] the proton transfer rate τ^{-1} can be obtained as a temporal decay of $C(t)$ at long time span.[39] The span will be changed on click from the tool GUI until finding a reasonable value of rate.

3.3.2.2. Hydronium-water local density field ρ

Using this fitting function will allow composing an input file through spatial correlations with respect to the hydronium 'NH_3O' label following

$$\rho_{0*\alpha}(r) = \frac{1}{4\pi r^2}\Big\langle \sum_i \delta\big(|r_{0*} - r_i^{\alpha}| - r\big)\Big\rangle, \tag{3}$$

where r_{0*} is the coordinate of the instantaneous hydronium molecular, and r_i^{α} is the coordinate of the i-th site of species α ($\alpha = \mathrm{O}, \mathrm{H}$). This input will be tested with neural networks until getting a smooth chart to present the locations of main important peaks at different *pivot* solvent density fields.

3.3.2.3. Diffusion coefficient of proton

The analyzing tool also provides a function to allow computing the diffusion coefficient of the proton D_{H^+} by following Einstein's relationship

$$D_{\mathrm{H}^+} = \frac{1}{6}\lim_{t\to\infty}\frac{\mathrm{d}\big\langle |r_{\mathrm{H}}(t) - r_{\mathrm{H}}(0)|^2\big\rangle}{\mathrm{d}t},$$

where

$$r_{\mathrm{H}}(t) = \sum_{i=1}^{N_{\mathrm{EVB}}} c_i^2 r_i^0(t)$$

is the proton coordinate in terms of the weighted sum of EVB states that participate in EVB Hamiltonian.[37] The use will simply compose the input file based on the data set (Time, Xp, Yp, Zp) to calculate the mean square displacement and then test this file with neural network until getting a smooth chart to find the slope $\mathrm{d}\langle |r_{\mathrm{H}}(t) - r_{\mathrm{H}}(0)|^2\rangle/(\mathrm{d}t)$ and D_{H^+}.

4. Running the EVB pattern recognition tool

The new tool could be easily run via its flexible GUI. "*Ptransfer*" would be the new name of the modified EVB code including its add-on tools. Figure 5 shows the main window of the tool. Screen monitor appears while running the EVB code. After setting the input file, at any time user can stop simulation. During the simulation *Ptransfer* will continue plotting patterns step by step. At the end, user can start data file resampling from "tool menu", to find pattern on charts and plot smooth charts as shown in Fig. 6.

From the GUI of the tool, user could start to train and test the output data of the EVB code. Herein, ANN network must

be set up according to the user needs. Choosing the number of neurons as the number of possible classes in the data would be strongly recommended. User needs just to enter historical data as inputs and results (desired outputs) by browsing the saved output file from any computer directory (C, D, E, …, etc.). If user wants a pattern recognizer, then he must use a column for each pattern in the desired outputs table as appeared in Fig. 7.

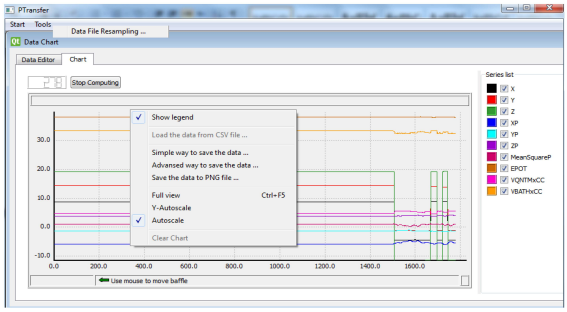


Fig. 5. (color online) Main window of *Ptransfer* .

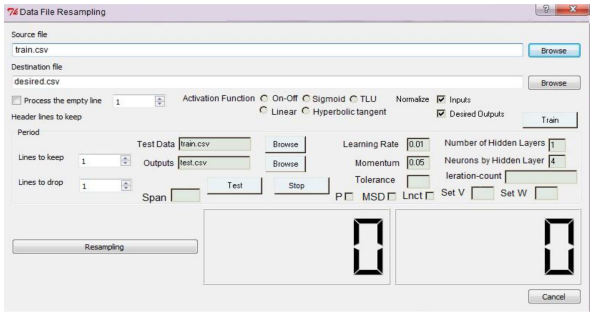


Fig. 6. (color online) The main window of the ANN application, which is the add-on tool of EVB code (data filtering and re-sampling, data editor including and file manipulation).

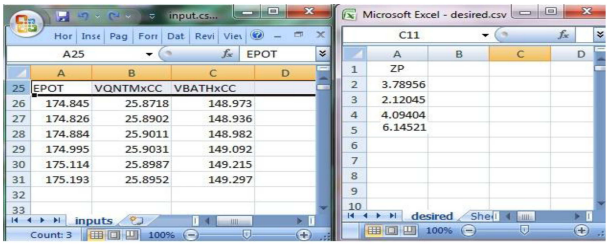


Fig. 7. (color online) Input data set, two files of training phase.

The user should use the "browse" button in the main screen to enter the two required input files to be trained after pressing "train": file1 (input.csv), file2 (output.csv). Data set consists of two inputs (I_1 and I_2) and one output (O). Here there are some re-sampled data from the computed results of EVB output. But user can set the values based on his case study just to learn the network. Once the neural network would be trained, user could use new sets of data (test) to predict their possible results (outputs) according to what has been learned in the training as presented in Fig. 8.

On the other hand, as patterns and relationships between numbers are often easier to recognize when being shown in a chart, this tool allows drawing charts simply by loading files from database with EVB output results to the main window of the tool on a right click, see Fig. 5. After re-sampling the required file of a large set of data, the user can set required fitting function ($\ln c(t)$, MSD, and density fields) on click, directly a new window will be open with presenting the required chart pattern to start to analyze data, for more accuracy user can change the ANN fitting parameter constant until getting a satisfied result as would be explained in the case study, see Fig. 3.

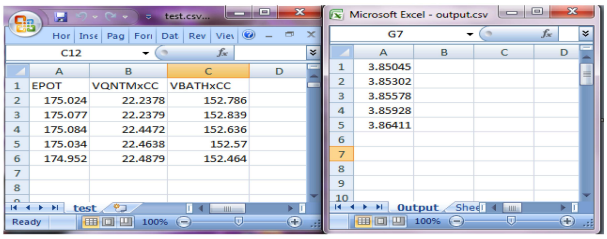


Fig. 8. (color online) Input files, for testing phase insert them then press the test.

5. Finding proton transfer chemical properties in aqueous environments based on the proposed method

This section presents the using of such a new tool for solving a real problem. in this situation this required expanding the example that appears in Fig. 2 and taking into account the tool structure diagram of Fig. 3 to see how it logically works and how the better output results can be obtained.

5.1. Physical case study

As is well known, confining environments, with characteristic lengths on a nanometer scale, are able to significantly change the equilibrium properties and dynamics associated with any reactive process in solution. In this regard, this case study will show how to provide the EVB code user with clear chart patterns to analyze the structural properties and dynamics of solvation of an excess proton located in aqueous environments of pure water, in order to find the most important chemical propriety so called proton transfer rate. Knowing that this example could be repeated using another queues environment, e.g., strong acids and ionic solutions confined in different geometries (varying from hydrophobic to hydrophilic nature).

5.2. Data statistics of model variables

This example is to run the EVB code for a long simulation (10^5 fs) of time, for evaluating the proton transfer rate in a bulk of pure water with 125 water molecules at a density of 1 g·cm^{-3} and three temperatures $T = 250$, 273, 298 K based on using the new tool. User must run the EVB code to generate the output files for the three temperatures. These output files will be stored in the new tool data base to be used as an input

of the ANN pattern recognitions tool, starting with composing the sample of input data with extracting and fitting the required data, simply on clicking the relevant icon from the main window of the tool. In this case $\ln C(t)$ will be the target. As we pointed out above, this statistical function could be defined as regularities in the output data of EVB in order to create the required chart patterns of proton transfer. The computed value of proton transfer rate in this case study is trained and tested several times by changing the number of neurons in hidden layer, thus for achieving more accuracy in the learning process: this number has been changed randomly from 1 neuron to 15 neurons using the main window after several runs: another attempt could be done with using other constants. This allows training the modified data with different initial values of the hydronium label values in order to reach a suitable average data set. Figure 9(a) shows three crowded chart patterns. Figure 9(b) presents the results obtained for the dynamics of proton transfer and their conversion to a simple chart, by using new computational tools. A comparison between the two figures shows that smooth chart in Fig. 9(b) is only a pattern of proton transferred "occurred", which presents a clear behavior with respect to time, which could be used to analyze the proton transfer instead of using the unsophisticated chart of Fig. 9(a). One additional feature is that the tool will allow drawing the three charts together with the experimental chart in single one chart as shown in Fig. 9(b).

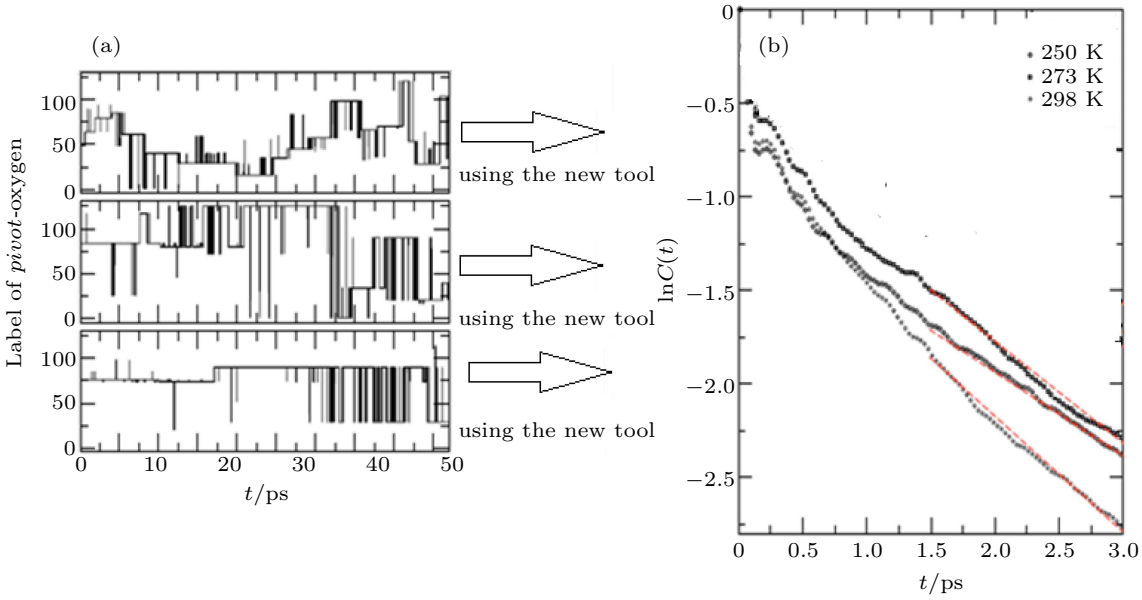


Fig. 9. (color online) Plots of logarithm of population relaxations of hydronium label (*pivot*) versus time at different values of water temperature $T = 250$, 273, 300 K and $\rho = 1.0$ g·cm^{-3}, where red dotted lines represent experimental tested data.

6. Results and discussion

Three-layer ANN with a tangent sigmoid transfer function at hidden layer and one linear transfer function at output layer has been used in the above case study to find the proton transfer rate. Eleven neurons have been used in the hidden layer. Furthermore, this is not a fixed ANN structure and the tool will simply allow constructing any ANN structure from the main window following the variables of any other case study (e.g., 20-layer ANN with another activation function tansh, ..., etc).

6.1. Optimization of the ANN structure

The optimal architecture of the ANN pattern recognition model in addition to its parameter variation is determined based on the minimum value of the mean square error (MSE) of the training as well as prediction set. MSE is defined as

$$\mathrm{MSE} = \frac{1}{N}\sum_{A=1}^{N} d_A^2 = \frac{1}{N}\sum_{A=1}^{N} (t_A - o_A)^2,$$

where N denotes the number of patterns, A the pattern index, t_A the A-th target value (experimental value), O_A the A-th output, d_A the difference between t_A and O_A. Figure 10 illustrates the relation between the neuron number at hidden layer and MSE for the MLP back propagation algorithm.

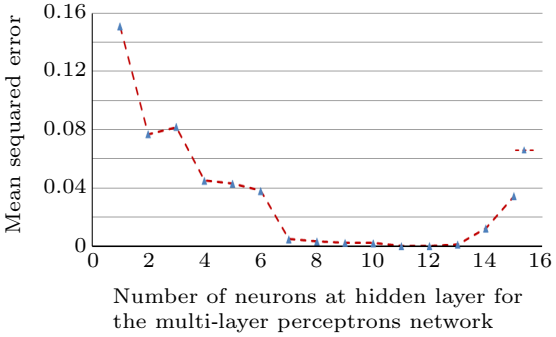


Fig. 10. (color online) Variation of the MSE with the number of neurons at hidden layer.

As a result, with the increasing of the number of neurons, the network presents several local minimum values in addi-

tion to different MSE values that are obtained for the training set. One neuron is used as an initial guess in the hidden layer for optimizing the ANN. The MSE of the network is much higher for 1 (MSE 0.151) and 2 (MSE 0.077) hidden neurons than those with 3(MSE 0.082), 4 (MSE 0.045), 5 (MSE 0.043), and 6 (MSE0.038). With 7 hidden neurons, the MSE decreases significantly from 0.038 to 0.0048. With the number of neurons further increasing from 7 to 11, the MSE gradually decreases. The MSE reaches its minimum value of (MSE 0.000117) with 11 hidden neurons. For this reason, the ANN containing 11 hidden neurons (MSE 0.000117) is chosen as the best case. When the number of neurons exceeds 11, the MSE shows a slight increase from 0.000117 to 0.0012 at 13 neurons. A sharp increase in the MSE results from the number of neurons further increasing from 13 to 15. This growth can be attributed to the characteristics of the MSE performance index and the input vector [A] used in this case study. Here, the initial weight plays an important role in learning but unfortunately no one could estimate the proper initial weight set and each attempt starts with the learning process and again uses different random initial weights. Finally when the differences between training error and validation error starts to increase, training stops after iterations 1000 for the MLP network. The flowchart of the MLP algorithm together with this optimal 3-layer ANN is shown in Fig. 3. On the other hand, a statistical analysis for estimating the relationships between input variables and network response between ANN graphical output of the network and the corresponding experimental data is performed as indicated in Fig. 9. The performance control of ANN graphical output is evaluated by estimating the correlation coefficient (R_2) which is defined as

$$R^2 = \frac{\sum_{A=1}^{N}(t_A - t_{\text{mean}})^2 - \sum_{A=1}^{N}(t_A - O_A)}{\sum_{A=1}^{N}(t_A - O_A)^2}^2,$$

where R denotes the correlation coefficient, N the number of patterns, A the index number of pattern, t_A the target value (experimental value), O_A the output of the A-th pattern which is proposed by ANN model. Therefore, test outputs show a very small deviation in proton transfer rate values from the experimental data with an average value of about $0.12(\pm 0.19)$ where the proton transfer rates at three different temperatures are found to be 0.45, 0.47, 0.50 ps^{-1}, obtained from a window between 1000 fs and 2000 fs, for 250, 273, and 298 K respectively. Therefore it can be seen that the proposed ANN model shows a good performance in the prediction of the experimental data based on R^2 values in Table 2. These results show good agreement with previous studies.[37]

Table 2. Model variables for computing proton transfer rate.

Variables	Range
Input layer [A]	
Simulation time [A_1]	$1 \sim 10^5$ fs
Temperatures [A_2]	$T = 250$, 273, 298 K
Correlation function of *pivot* [A_3]	1–300; data modification to reduce the size of data set of simulation time from 10^5 to 300 based on the number of tagged water molecules (i.e., 49, 64, 49, 118, ...).
Density [A_4]	0.0333 Å^{-3}
Number of water molecules [A_5]	125 to 400
Output layer	
Proton transfer rate could be any known experimental value	

6.2. Sensitivity analysis

In this work, a sensitivity analysis is conducted for determining the degree of effectiveness of a variable using the proposed ANN pattern recognition model. In this analysis, performance evaluations of various possible combinations of variables are investigated. Consequently, performances of the groups of (1, 2, 3, 4, 5) variables are tested here based on the optimal ANN structure by using the MLP with 11 hidden neurons. The groups of input vectors are defined in [A_1, A_2, A_3, A_4, and A_5] arrays as shown in Fig. 3 and Table 2. The groups vary from groups of 1 to 5 variables with all expected combination probabilities, such as one-variable group (e.g., examples, A_1, A_2, A_3, A_4, A_5); two-variable groups, (e.g., A_1+A_2, A_1+A_3, A_1+A_4, A_1+A_5, A_2+A_3, A_2+A_4, A_2+A_5, A_3+A_4, A_3+A_5, A_4+A_5); three-variable groups (e.g., $A_1+A_2+A_3$, $A_1+A_2+A_4$, $A_1+A_2+A_5$, $A_1+A_3+A_4$, $A_1+A_4+A_5$, $A_2+A_3+A_4$, $A_2+A_3+A_4$, $A_2+A_3+A_5$, $A_2+A_4+A_5$, $A_3+A_4+A_5$), four-variable groups (e.g., $A_1+A_2+A_3+A_4$, $A_1+A_2+A_3+A_5$, $A_1+A_2+A_4+A_5$, $A_1+A_3+A_4+A_5$, $A_2+A_3+A_4+A_5$), and finally five-variable groups (e.g., $A_1+A_2+A_3+A_4+A_5$). Then the best results of the performance evaluation of these 30 combinations are summarized in Table 3.

Table 3. The best group performances according to the number of parameters. Gradient is a formula used to update the search direction in internal parameter space. For more details see Ref. [33].

Combination	MSE	R^2	Iteration	Gradient	Best linear equation
A_2	82.456	0.832	1000	2.923×10^{-9}	$Y = 0.69X + 23.6$
A_2+A_3	0.1174	0.858	800	3.122×10^{-14}	$Y = 0.79X + 9.3$
$A_1+A_2+A_3$	0.05631	0.919	780	3.831×10^{-14}	$Y = 0.71X + 15.3$
$A_1+A_2+A_3+A_5$	0.02921	0.921	900	6.321×10^{-8}	$Y = 0.84X + 11.4$
$A_1+A_2+A_3+A_4+A_5$	0.000011671	0.949	1000	0.398×10^{-10}	$Y = 0.97X + 7.8$

Performance evaluation of combinations of input variables shows good agreement between ANN output and experimental data as a function of temperature [A_2] for the one-variable group that gives the lowest MSE. The temperature A_2 and correlation function of *pivot* A_3 is the best combination of two variables to affect the proton transfer rate computed value, it was very close to the experimental value with a minimum MSE in the two-variable group, which is determined to be 0.1174, i.e., MSE = 0.1174. As for the constant density it has no effect on the results, in which predicting the proton transfer rate will not be possible based only on the water density A_4, but also it will have a clear effect on the 5-variable group. The minimum MSE in the four-variable group ($A_1+A_2+A_3+A_5$) is determined to be 0.02921 with no effect of density A_4. For 5 operating variables ($A_1+A_2+A_3+A_4+A_5$), the proposed ANN pattern recognition model shows an accurate and effective prediction of the experimental data with a very satisfactory correlation coefficient of 0.949 with a very small MSR of 0.000011671. The sensitivity analysis shows that MSE value decreases as the number of variables used in the ANN model increases. The relative increase in the performance due to inclusions of A_2 (temperature) is found to be larger than due to the contributions of others. From the above discussion it follows that a simulation based on the ANN model can provide a further contribution to develop a better understanding of the dynamic behavior of proton transfer process where some phenomena still cannot be explained in detail but in some manners the effects of these variables will explain the physical meanings of the phenomena. These results show that they are in good agreement with literature results such as in Refs. [33] and [35], which confirms what is already known about MLP network behavior, subsequently, this will confirm the fact that a pattern recognition problem of proton transfer could be well solved by using the feed-forward back-propagation neural network approach. In this regard, it is noteworthy to mention that this phenomenon has been the main topic of many previous studies based on developing such ANN applications,[33,35–44] but each study has its particular objectives and it targets some specific chemical characteristics, usually other than those we have been adopted here (e.g, proton transfer rate) but all of these studies including the current study, have agreed on the importance in using the ANN neural network, for solving the problem of nonlinearity relationship between input variables and the required output. Therefore, the proposed tool is expected to be used as a supplementary manner in analyzing the proton transfer phenomena and would help researchers make their decisions toward this phenomenon using the EVB code based on automatical chart fitting and drawing of the large output file in one workstation via a flexible GUI. Furthermore, any user can use this tool simply by following its manual, without having to learn the theory or programming languages to start to analyze his data and reported scientific findings.

7. Conclusion

This study successfully presents a flexible utility program to simplify the study of proton transfer phenomena via using the EVB code. This utility program consists of a flexible GUI in addition to a pattern recognition tool supported with an easy-to-use data base system. The tool could be run in several operating systems (Windows, Unix, LINUX, MacIntosh) with no need to install supplementary and computer configuration software. The tool can simplify plotting and re-sampling as well as data mining via the EVB code output files, and now it can be used as a supplementary material for analyzing the proton transfer phenomenon in several aqueous environments and accessing its deeper physical knowledge. The new linked tool defined a pattern recognition Python tool as a deployed system for the classification of proton transfer data patterns as well as categorizing them into predefined set of classes based on the EVB code generated output files to be trained and tested using the feed forward back propagation algorithm that would repeat the procedure of weight adjustments until the error would be reduced to a negligible level in the error tolerance range. Therefore, using feedback information obtained from the difference between the actual and the desired results could increase the performance and the efficiency of the network. This information would then be used for adjusting the interconnections between the neurons at the input layer for matching the actual result with the desired one.

Acknowledgment

The authors thank Dr. Steve Jones, Scientific Advisor of the Canon Foundation for Scientific Research (7200 The Quorum, Oxford Business Park, Oxford OX4 2JZ, England). Canon Foundation for Scientific Research funded the UPC 2013 tuition fees of the corresponding author during her writing this article.

References

[1] Jain A, Duin R and Mao J 2000 *IEEE Trans. PAMI* **22** 4
[2] Neimark Y 2003 *Mathematical Models in Natural Science and Engineering* (Berlin: Springer) p. 25
[3] Callan R 1999 *Essence of Neural Networks* (London: Prentice Hall Europe) p. 223
[4] Abreu P and Lesiak T 1992 *Phys. Lett. B* **295** 382
[5] Denby B 1999 *Comput. Phys. Commun.* **119** 219
[6] Abachi S , Abbott B, Abolins M, Acharya,B, Adam I, Adams D, Adams M, Ahn S, Aihara H, Alves G, Amidi E, Amos N, Anderson E, Astur R, Baarmand M, Baden A, Balamurali V and Ba J 1997 *Phys. Rev. Lett.* **79** 1197
[7] Abbott B, Abolins M, Bannanje Sripath Acharya and Adam I 1997 *Phys. Rev. Lett.* **79** 4321
[8] Zhu M H, Liu L G, Xu A A and Ma T 2008 *Chin. Phys. Lett.* **25** 3942
[9] Zhou Z B, Wang S F and Luo J 2005 *Chin. Phys. Lett.* **22** 1614
[10] Gao X F, Wan B K and Yang C M 2003 *Chin. Phys. Lett.* **20** 774
[11] Yu Q Z, Li Y T, Zhang J, Zheng J, Li H M, Peng X Y and Li K 2004 *Chin. Phys. Lett.* **21** 874

[12] Sun L S, Kang X Y, Lin L X and Zhang Q 2011 *Chin. Phys. B* **20** 120507
[13] Amato F, López1A, Peña-Méndez E, Vaňhara P, Hamp A and Havel J 2013 *J. Appl. Biomed.* **11** 47
[14] Hynes J, Klinman J, Limbach H and Schowen R 2007 *Hydrogen Transfer Reactions* (Weinheim: Wiley VCH) p. 50
[15] Gorb L and Leszczynski J 1998 *J. Am. Chem. Soc.* **120** 5024
[16] Banerjee C, Ghatak C, Mandal S, Ghosh S, Kuchlyan J and Sarkar N 2013 *J. Phys. Chem. B* **117** 6906
[17] Lukeman M and Wan P 2001 *Chem. Commun.* **11** 1004
[18] Ma J, Zhao J, Yang P, Huang D, Zhang C and Li Q 2012 *Chem. Commun.* **48** 9720
[19] Golan A, Bravaya K, Kudirka R, Kostko O, Leone S, Krylov A and Ahmed M 2012 *Nat. Chem.* **4** 323
[20] Ramette R 1989 *J. Chem. Educ.* **66** 830
[21] Car R and Parrinello M 1985 *Phys. Rev. Lett.* **55** 2471
[22] Kornyshev A, Kuznetsov A, Spohr E and Ulstrup J 2003 *J. Phys. Chem. B* **107** 3351
[23] Cuma M, Schmitt U and Voth G 2000 *Chem. Phys.* **258** 187
[24] Warshel A and Weiss R 1980 *J. Am. Chem. Soc.* **102** 6218
[25] Jelen B 2011 *Charts and Graphs: Microsoft Excel 2010*, (Indiana: Indianapolis) p. 25
[26] Zundel G and Metzger H Z 1968 *Phys. Chem.* **244** 456
[27] Eigen M and de Maeyer L 1958 *Proc. Royal Soc. London A* **247** 505
[28] Duda R, Hart P and Stork D 2001 *Pattern Classification* (New York: John Wiley and Sons) p. 72
[29] Tahat A, Marti J, Tahat K and Khwaldeh A 2013 *Chin. Phys. B* **22** 048901
[30] Bishop C M 1995 *Neural Networks for Pattern Recognition* (Oxford: Oxford University Press) p. 80
[31] Ripley B 1996 *Pattern Recognition and Neural Networks* (Cambridge: Cambridge University press) p. 200
[32] Haykim S 1994 *Neural Networks: A Comprehensive Foundation* (New York: Macmillan College Publishing Company and Maxwell Macmillan International) p. 113
[33] Christopher M 1995 *Neural Networks for Pattern Recognition* (Oxford: Clarendon Press) p. 301
[34] Schalkoff R 1992 *Pattern Recognition: Statistical, Syntactic and Neural Approaches* (New York: John Wiley and Sons) p. 140
[35] Aleksander I and Morton H 1995 *An Introduction to Neural Computing* (London: International Thomson Computer Press) p. 97
[36] Lippmann P 1987 *IEEE ASSP Magazine* **3** 4
[37] Laria D, Martí J and Guàrdia E 2004 *J. Am. Chem. Soc.* **126** 2125
[38] Chandler D 1987 *Introduction to Modern Statistical Physics* (New York: Oxford University Press) p. 200
[39] Semino R and Laria D 2012 *Chem. Phys.* **136** 194503
[40] Chitra S 1992 *Chem. Eng. Prog.* **89** 44
[41] Freeman J and Skapura D 1991 *Neural Networks: Algorithms, Applications and Programming Techniques* (Redwood City: Addison Wesley Longman) p. 300
[42] Karray F and de Silva C 2004 *Soft Computing and Intelligent Systems Design — Theory, Tools and Applications* (Redwood City: Addison Wesley) p. 43
[43] Rezaei M and Mohseni M 2013 *J. Fuel Cell Sci. Technol.* **10** 035001
[44] Fatemi M H and Moghaddam M R 2012 *J. Mass Spectrom.* **47** 574