



6-10-2014

## A Neural Network Structure for Prediction of Chemical Agent Fate

Homayun K. Navaz

N. Kehtarnavaz

Zoran Jovic

Follow this and additional works at: [https://digitalcommons.kettering.edu/mech\\_eng\\_conference](https://digitalcommons.kettering.edu/mech_eng_conference)



Part of the [Mechanical Engineering Commons](#)

---

# PROCEEDINGS OF SPIE

[SPIDigitalLibrary.org/conference-proceedings-of-spie](http://SPIDigitalLibrary.org/conference-proceedings-of-spie)

## A neural network structure for prediction of chemical agent fate

Navaz, H., Kehtarnavaz, N., Jovic, Zoran

H. K. Navaz, N. Kehtarnavaz, Zoran Jovic, "A neural network structure for prediction of chemical agent fate," Proc. SPIE 9073, Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XV, 907309 (10 June 2014); doi: 10.1117/12.2048593

**SPIE.**

Event: SPIE Defense + Security, 2014, Baltimore, Maryland, United States

# A neural network structure for prediction of chemical agent fate

H. K. Navaz, Department of Mechanical Engineering, Kettering University, Flint, Michigan  
N. Kehtarnavaz, Department of Electrical Engineering, UT Dallas, Dallas, Texas  
Zoran Jovic, Research Engineer, Kettering University, Flint, Michigan

## ABSTRACT

This work presents the development of a multi-input, multi-output neural network structure to predict the time dependent concentration of chemical agents as they participate in chemical reaction with environmental substrates or moisture content within these substrates. The neural network prediction is based on a computationally or experimentally produced database that includes the concentration of all chemicals presents (reactants and products) as a function of the chemical agent droplet size, wind speed, temperature, and turbulence. The utilization of this prediction structure is made user-friendly via an easy-to-use graphical user interface. Furthermore, upon the knowledge of the time-varying environmental parameters (wind speed and temperature that are usually recorded and available), the time varying concentration of all chemicals can be predicted almost instantaneously by recalling the previously trained network. The network prediction was compared with actual open air test data and the results were found to match.

## 1. INTRODUCTION

The spread of chemical warfare agents (CWA's) in the environment poses a great threat to warfighters as well as the general public. This threat becomes more serious if their release takes place with malicious intentions. The field commanders and/or first responders have to make quick but difficult decisions that have direct impacts on the lives of the soldiers and the citizens. It will be preferred that these decisions to be based on scientific analysis and data regarding the spread and distribution of these CWA's into the environmental substrates or atmosphere. When a chemical agent is disseminated onto environmental substrates several processes occur simultaneously. These processes are surface evaporation, capillary transport through the substrate, evaporation inside the pores, chemical reaction with the substrate and/or other pre-existing chemicals inside the pores, especially moisture and water, and adsorption/desorption processes.

These processes take place simultaneously and coupled with all the existing cross effects therefore obtaining a solution becomes very challenging. It is evident that the more detailed solution requires more computation time, and producing credible results or outcomes will be more time intensive and complex compromising the practicality aspects of the analysis. These two issues have led the CHEM/BIO (Chemical and Biological Defense) community to rely on experimental data, curve fitting the data (or some regression model) to address the practicality issue given that the actual real conditions lie within the data set. In spite of the simplicity and immediate implementation of this approach, it is not general, it has limited capability, and also costly. Note that performing experiments with actual CHEM/BIO agents under real and actual environmental conditions (wind speed, temperature, etc.) may not be readily possible or feasible.

The complexity of the problem and practical aspects of any useful tool prompted us to tackle this problem. It is evident that during the first phase, a robust and "accurate" computational model that couples all the physiochemical processes needed to be developed without any precondition of required computational time. This step was completed with government investment and the model can serve as a workhorse to produce any outcome based on computational fluid dynamics (CFD) modeling in any multi-phase multi-species porous media domain. This is documented in References [1] through [5] and also summarized in another paper being presented in this conference. Any simulation with this model although very useful and informative, requires excessive computer time therefore eliminating the practicality aspect of the model. This work is intended to address the transition of this model to a "practical tool". This practical tool is an artificial neural network program (ANN) that uses the database generated by the model and it is a temporal and spatial mapping of many outcomes based on the real time environmental data (wind and temperature) and conditions that it acquires through an interface. Figure 1 is a representation of the appropriate response tool (ART) that was developed in the spirit of providing the CHEM/BIO community with a "practical tool".

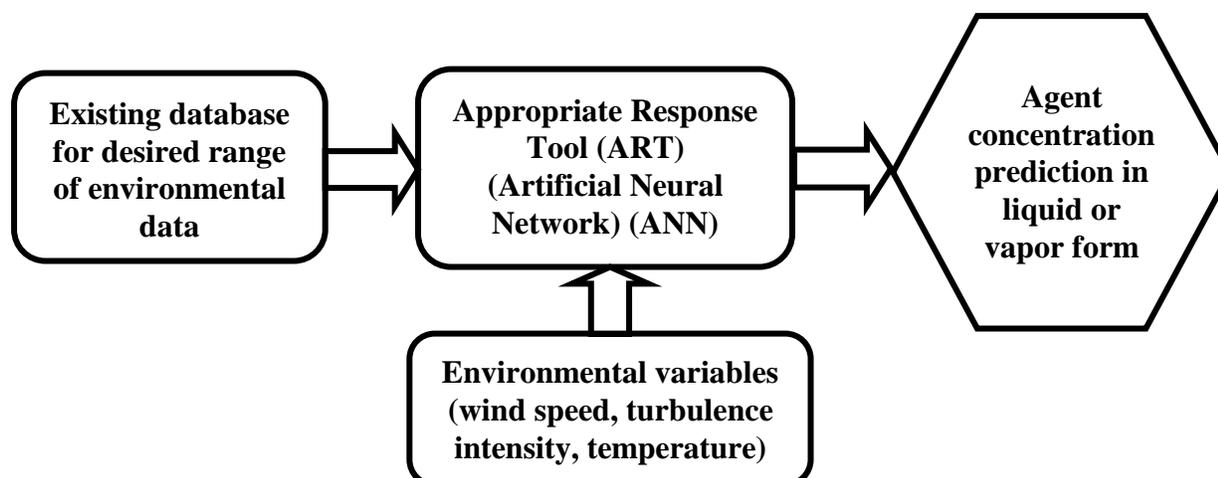


Figure 1. Schematic of the ANN functionality.

The database generated by the computational model consists of the agent (or any chemical) concentration predictions under variable inputs. The inputs that can affect the outcome are the initial chemicals mass (or droplet size, if any), wind speed, turbulence intensity, and temperature. The temperature affects the evaporation and reaction rate, the wind speed and turbulence intensity affects the surface evaporation or convective mass transfer at the surface. A minimum and maximum for each input is considered and the range is divided into several equally spaced increments. The higher the number of increments, the more accuracy can be expected. Generally, 5 droplet sizes, three turbulence intensity levels, five temperature values, and four wind speed levels are found to be sufficient for a typical problem. One input is changed at a time. For instance for the given intervals, 300 permutations are required to fill the entire space. For each case, a different output (agent or species concentration in liquid, vapor, or solid form) is expected. These 300 files form our database. Lower and higher resolution solutions can be obtained based on the required accuracy. It should be emphasized that all these runs are done off line and the database is fed into the ANN for training purpose. The environmental variables are measured with thermocouple and hot wire anemometry to specify the temperature, wind speed, and turbulence intensity in real time. A data acquisition system communicates this information to ANN at specified time intervals. This paper focuses on the neural network and the output components of Figure 1.

Artificial Neural Network (ANN) models were first introduced by [6], as was explained by [7]. ANN models are algorithms that predict an outcome variable based on the values of some independent variables. A typical ANN model learns from examples and processes the information through special training algorithms that are developed based on learning rules [8]. The ANN is a computer program that is capable of learning the patterns of the change of dependent variable(s) with respect to the independent variables via training samples. It resembles biological neural nets in two ways: (1) Knowledge is acquired through learning (training) processes, and (2) Knowledge is stored via inter-neuron connection strengths (weights). While neural networks have been used to various capacities within food and bioprocess technologies, such as the work of [9], [10], and [11], our focus will be on the application of neural network towards solving large scale problem for biosecurity purpose.

## 2. ARTIFICIAL NEURAL NETWORK PROGRAM DEVELOPMENT

ANNs provide an alternative paradigm or mechanism for performing regression or curve fitting as compared to classical regression methods. In other words, ANNs denote a tool that achieves an implementation of regression. The classical regression approaches are linear whereas an ANN is a generalization of regression to non-linear systems. Many papers in various fields have used both regression and ANN for approximation applications. In essence, the main attribute of utilizing this alternative paradigm is that when utilizing ANN, there is no need to assume an underlying distribution of data and ANN can be applied to multivariate non-linear problems. Furthermore, if the output data is spread over a wide range as input variables are altered, clustering can be applied first for the ANN to more accurately predict the outcome.

ANN models consist of at least one input layer, at least one hidden layer, and at least one output layer of nodes or processing elements. The first layer consists of  $n$  nodes corresponding to  $n$  input variables  $(x_0, \dots, x_n)$ , second layer (hidden layer) consists of  $m$  nodes, and finally the output layer consists of  $r$  nodes corresponding to  $r$  output variables  $(y_1, \dots, y_r)$ . The most commonly used ANN model is the feedforward backpropagation model. In this model, every output variable  $(y_k)$  is a function of all the nodes in the hidden layer, and each node in the hidden layer is a function of the input variables  $(x_0, \dots, x_n)$ . No interaction takes place between the nodes of the same layer. The hidden layer(s) allows the model to handle the nonlinearity and complexity of the relationship between the inputs and the outputs. An ANN model, outlined in Figure 2, estimates an output vector  $(\hat{Y})$  as close as possible to the actual output vector  $(Y)$  based on an input signal vector  $X$ . The node activation values get passed to the hidden layer using the weights  $(w_{ji})$ , associated with the hidden layer nodes and then passed to the output layer using the weights associated with the output layer nodes  $(v_{kj})$ . The numerical simulations can be performed off-line and the results can be used to train the ANN program. The ANN program constitutes a simple-to-use and extremely practical tool.

### 3. NEURAL NETWORK TRAINING

The next step is to train a feed-forward neural network by using the data with outliers removed. For this purpose, a single layer network is used with  $Nh$  number of hidden layer nodes. The input data  $(X)$  are fed into the network as a  $4-D$  vector and the curve fit parameters  $(Y)$  is considered to be the desired output.

For training, the classical back-propagation training is used, with a momentum and adaptive learning rate and the Levenberg–Marquardt (LM) training algorithm, [12], and [13]. The parameters that can be adjusted are the learning rate, input-to-hidden layer transfer function, ITF (log sigmoid, tan sigmoid), hidden-to-output transfer function, OTF (tan sigmoid, linear), number of epochs, number of steps to stop the training if error on the validation set increases, percentage of the data to divide randomly for training, testing and validation. Extensive experiments of neural network training parameters were carried out and it was found that the best results corresponded to the LM algorithm with a network of  $Nh=25$  hidden layers, a learning rate of  $0.05$ , ITF as tan sigmoid and OTF as linear and with the training, testing and validation percentages of *80 percent, 10 percent and 10 percent*, respectively.

The weights of the input to the hidden layer are saved as  $W$ , the weights of the hidden layer to the o/p layer as  $V$ , and the respective bias vectors as  $bl$  and  $bo$ . Figure 3 shows a typical training error.

### 4. NEURAL NETWORK PREDICTIONS FOR REAL APPLICATIONS

We have chosen two test cases for which a database was built and the neural network predictions are assessed. The first test case deals with the evaporation of HD (mustard nerve agent) on a non-permeable surface. The variable inputs are the initial droplet size, wind speed, turbulence intensity, and temperature. The evaporation rate varies as a function of all these inputs. Three droplet size, five temperature increments from  $15^\circ\text{C}$  to  $55^\circ\text{C}$ , three levels of turbulence intensity from 0% to 6%, six different wind speed from 0 to 15 m/s generate 270 cases for which the output (remaining mass as a function of time) was generated by the computational model [4]. Figure 4 compares the network prediction with one of the actual outputs (normalized remaining mass of HD). The comparison looks acceptable for any practical applications. A better resolution can get the initial point closer to one, but since the initial release is assumed to be known, the prediction for later times become more important. This prediction is very close to the actual data.

The second test case is the hydrolysis reaction of VX nerve agent in wet sand. The temperature, initial mass of VX, and sand saturation were the inputs and the amount of remaining VX mass and one of the products that is initially produced and then reacts with moisture were taken to be the output to be presented. There are several reaction products (EMPA,  $\text{VXH}^+$ , and EMPT) as discussed in Reference [4]. However, we have presented the output for two of these species VX and  $\text{VXH}^+$ .

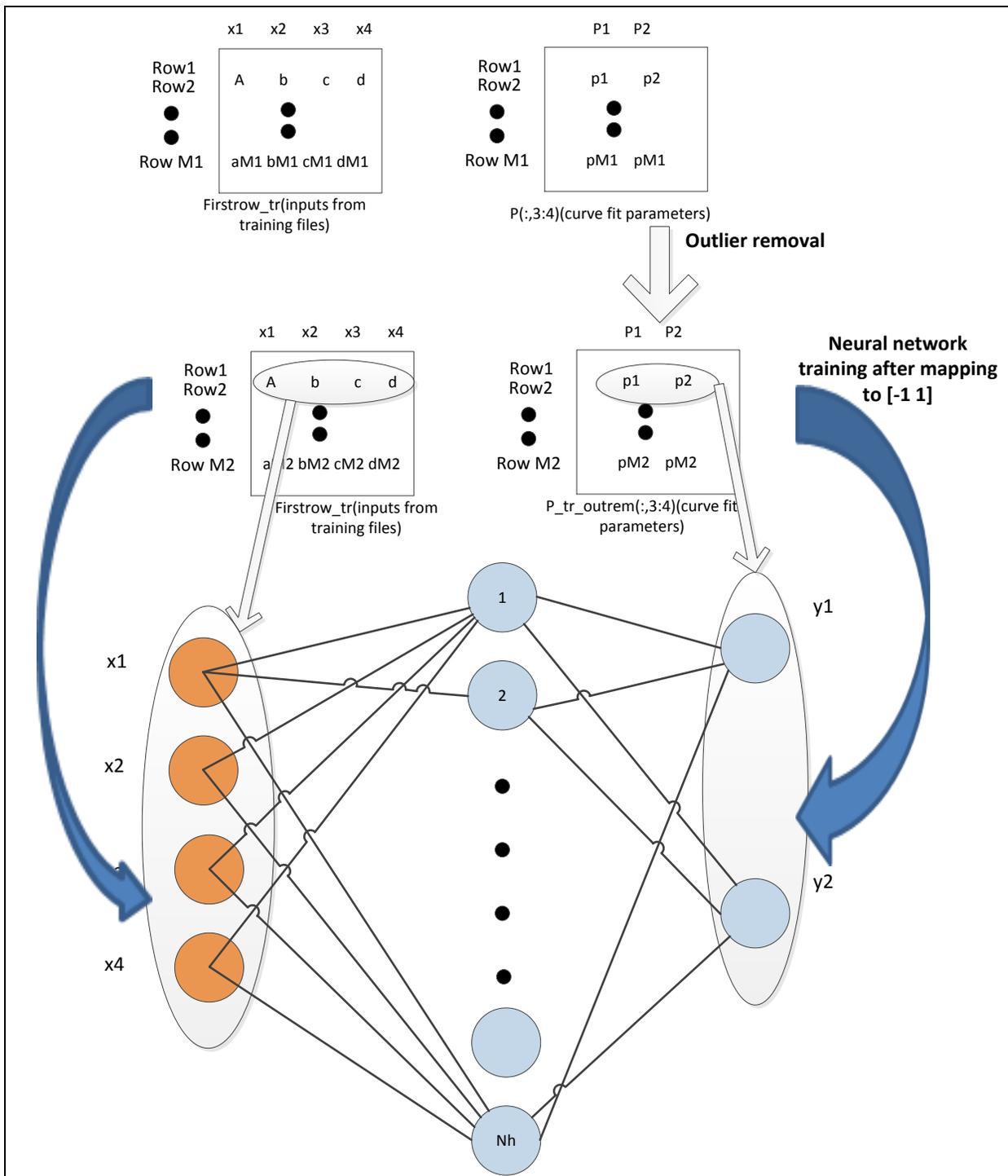


Figure 2. Outlier removal and neural network training.

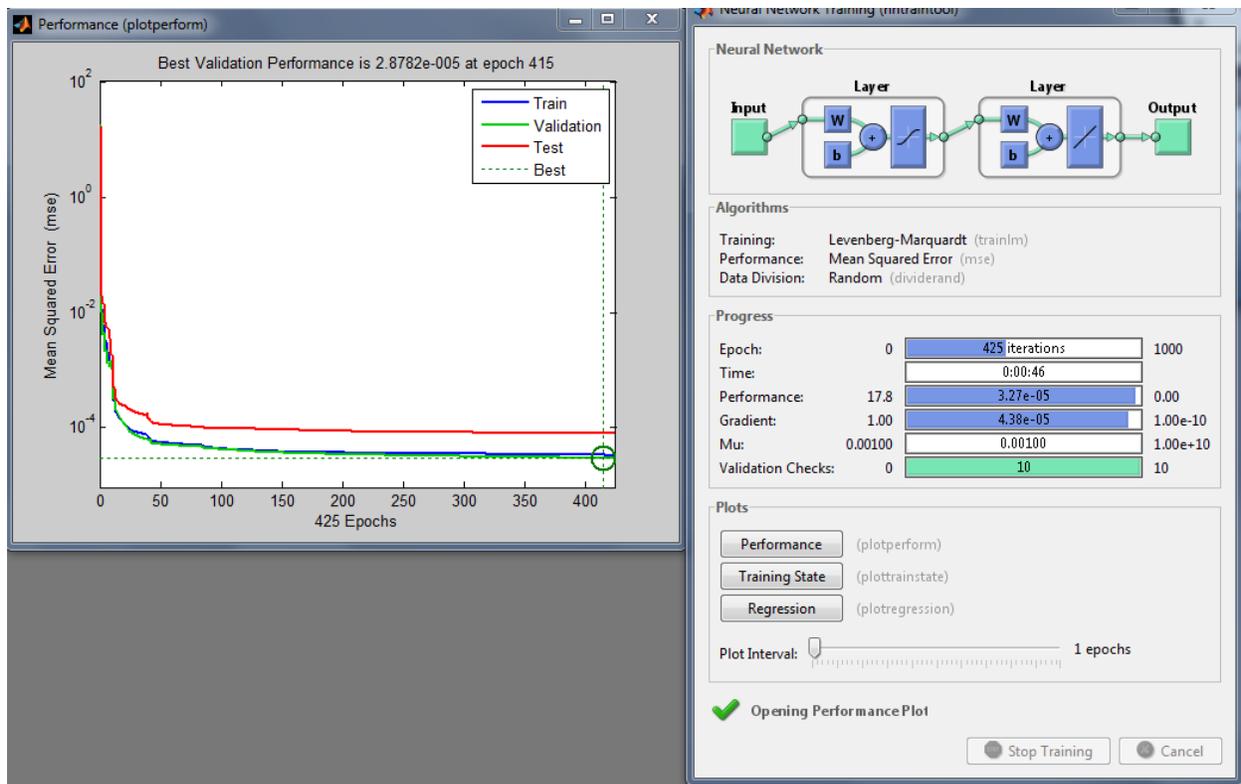


Figure 3. Neural network training error function.

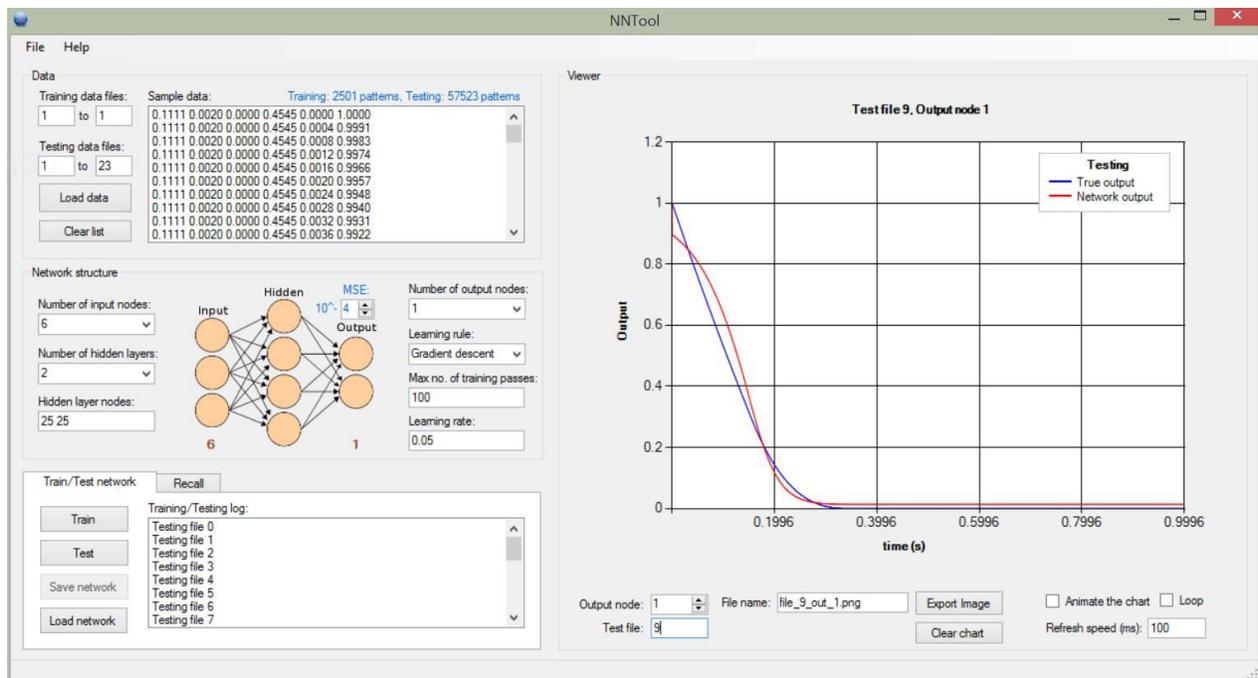


Figure 4. Network prediction for a sample HD evaporation.

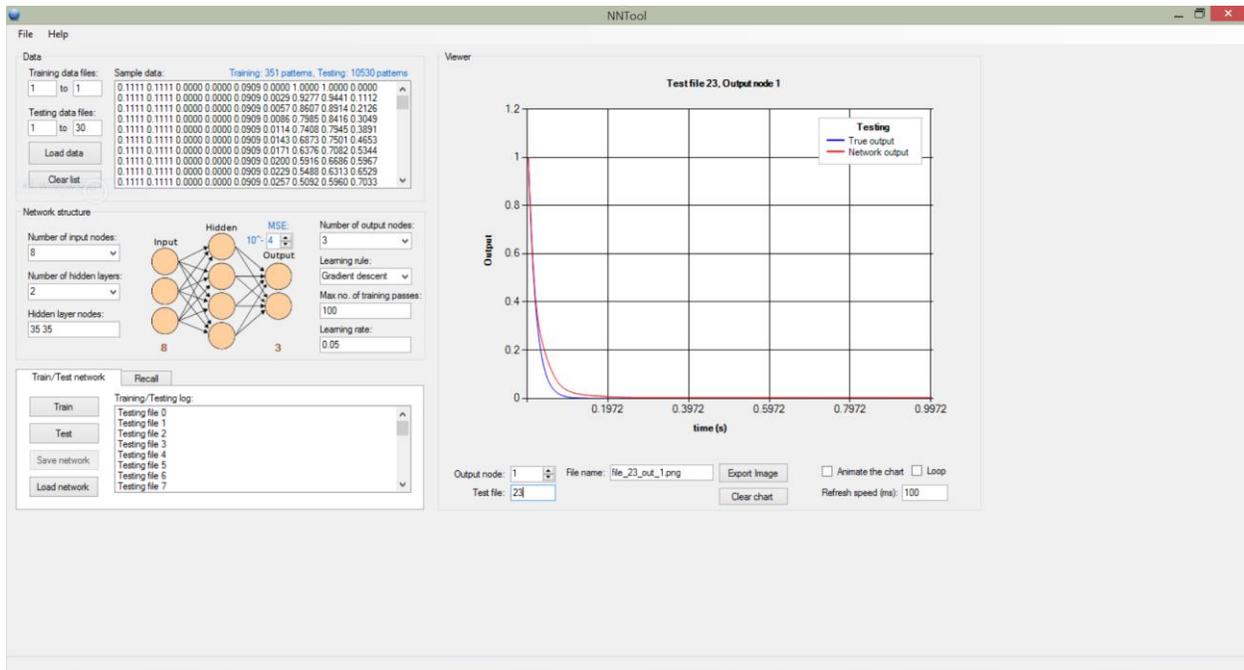


Figure 5. Network prediction for remaining VX normalized mass.

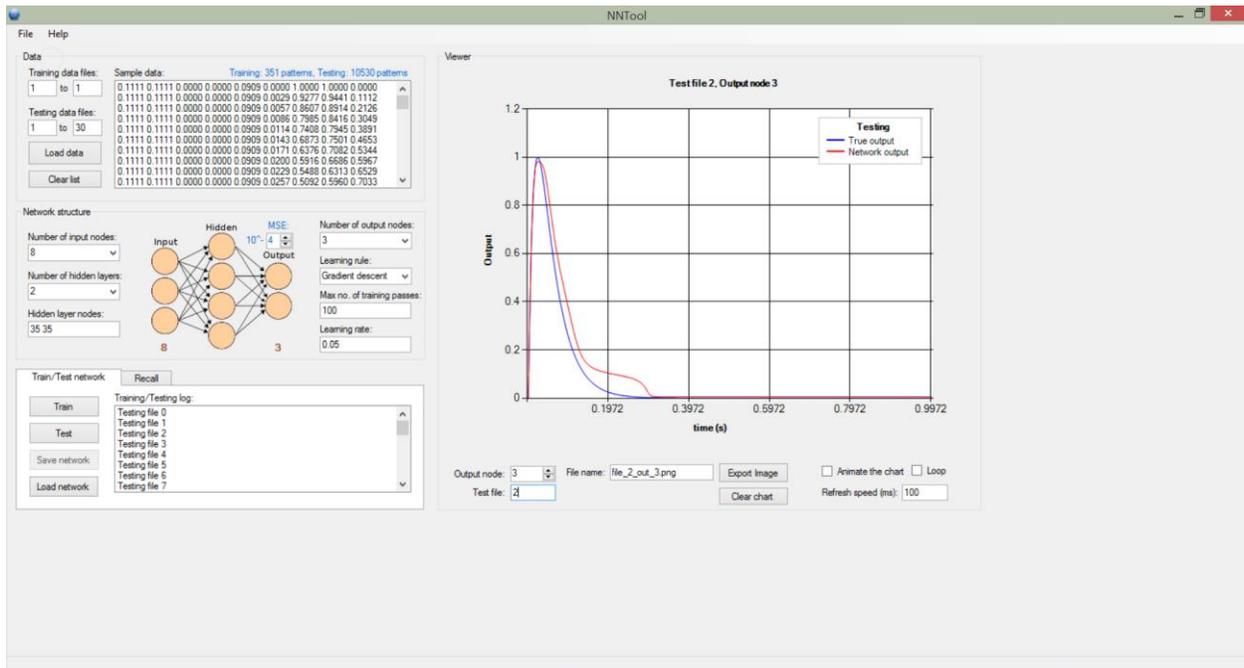


Figure 6. VXH+ normalized mass – Comparison between network and data.

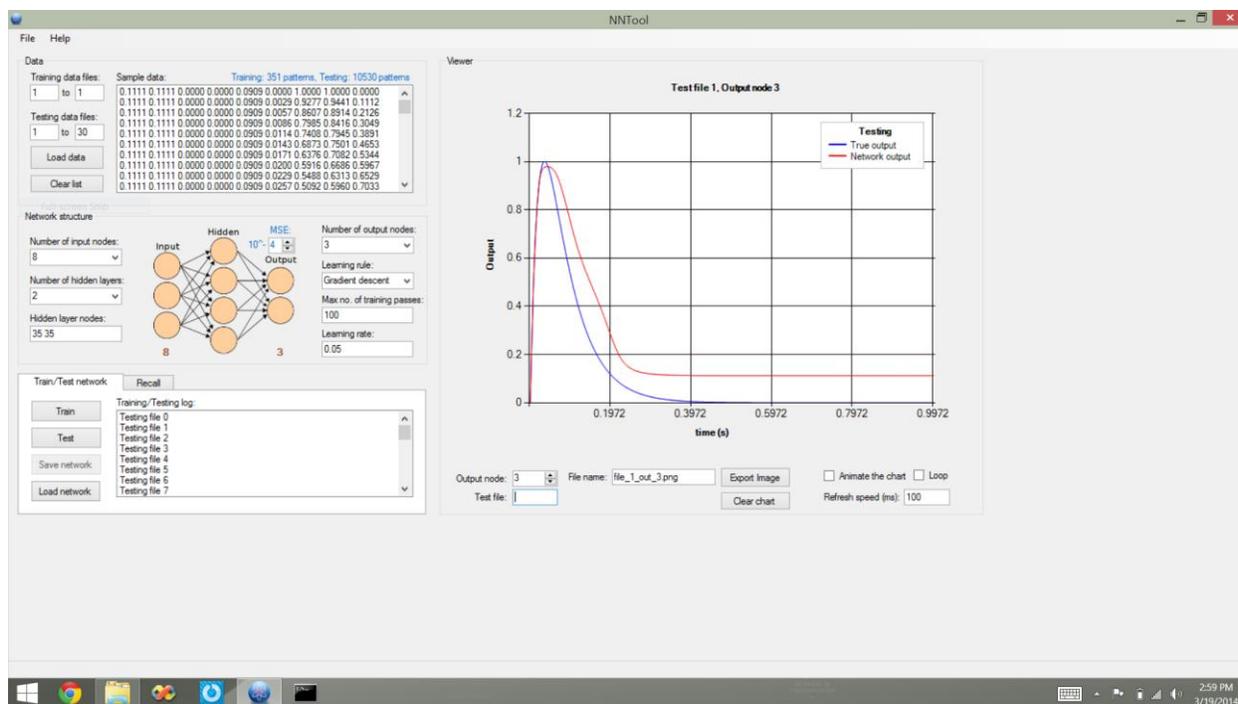


Figure 7.  $VXH^+$  normalized mass – Comparison between network and data.

Figure 5 compares the network prediction and data (generated by validated model) for the remaining VX. The comparison is excellent. Figures 6 and 7 compare the model and network predictions for  $VXH^+$  that goes through a more complex physiochemical process. It is initially produced as a product and after certain point it reacts and gradually depletes into other species. Figure 6 prediction is for higher temperature where the reaction occurs faster. It should be mentioned that due to the long computation time to generate the database for VX, only 27 cases (3 levels for each input variable) were generated. Considering the very low resolution of the database, the comparison seems to be adequate for practical applications.

## 5. CONCLUSIONS

It is demonstrated that the use of artificial neural network is a viable tool for transitioning difficult and computationally intensive CHEM/BIO modeling tools to a practical application that will be useful to commanders and first responders in the field. The software and data acquisition system has also been developed to interface with the network for real time output as the environmental variables (wind speed and temperature) change in time. Furthermore, based on real time weather input and output the threat level at later times can also be predicted.

## ACKNOWLEDGEMENTS

This work was performed with financial supported from the Edgewood Chemical and Biological Center (ECBC) and Defense Threat Reduction Agency.

## REFERENCES

- [1] Navaz, H. K., Zand, A., Dang, A. L., Gat, A., Gharib, M., Atkinson, T., Nowakowski, Kamensky, K., and Z. Jovic, (2014), Predictive Model for Assessment of Chemicals on and in Surfaces vs. Chemicals Available for Contact and Transport – Volume I - Theory and Analysis. Final Report prepared for Defense Threat Reduction Agency (DTRA).
- [2] Atkinson, T., Navaz, H. K., Zand, A., Jackson, J., Nowakowski, A., (2013), "Fate of a Sessile Droplet Absorbed into a Porous Surface Experiencing Chemical Degradation.," Submitted to the *AICHE Journal*.
- [3] Navaz, H.K., Zand, A., Atkinson, T., Gat, A., Nowakowski, A., and S. Paikoff, (2013), "Contact Dynamic Modeling of a Liquid Droplet between Two Approaching Porous Materials," Accepted for publication in *AICHE J.*
- [4] Navaz, H. K., Zand, A., Atkinson, T., Nowakowski, A. Kiple, J., Kamensky, K., and Z. Jovic, (2013), "Agent Fate Modeling-COMCAD Theory and Analysis, Code Structure, and Users Guide, Volume I, II, and III), Final Report prepared for Edgewood Chemical and Biological Center (ECBC).
- [5] Navaz, H. K., Zand, A., Markicevic, B., Herman, M., Atkinson, T., Li, Nowakowski, A., Gheresus II, P., Rothstein, M., Kiple, J., V. Middleton, (2011), Agent Fate Model Integration, Final Report, Prepared for Edgewood Chemical and Biological Center (ECBC).
- [6] McCulloch W-S & Pitts, W-H (1943) A logical calculus of the ideas immanent in nervous activity. **Bulletin of Mathematical Biophysics**, Vol. 5, pp 115-133.
- [7] Cheng B. & Titterington D-M (1994) Neural Networks: A Review from a Statistical Perspective. *Statistical Science*, Vol. 9, No. 1, pp 2-54.
- [8] Hajmeer M. N. & Basheer I-A (2003) Review of Microbiological Modeling Techniques for Meat Industry. American Society of Agricultural and Biological Engineers, ASABE Library.
- [9] Martins R-C, Lopes V-V, Vicente A-A & Teixeira J-A (2008) Computational Shelf-Life Dating: Complex Systems Approaches to Food Quality and Safety. *Journal of Food and Biotechnology Technology*, 1(3).
- [10] Boyaci I-K, Sumnu G & Sakıyan O (2008) Estimation of Dielectric Properties of Cakes Based on Porosity, Moisture Content, and Formulations Using Statistical Methods and Artificial Neural Networks. *Journal of Food and Biotechnology Technology*, 1(3). DOI 10.1007/s11947-008-0064-z
- [11] Patnaik P-R (2008) Intelligent Models of the Quantitative Behavior of Microbial Systems. *Journal of Food and Biotechnology Technology*. DOI 10.1007/s11947-008-0112-8.
- [12] Kenneth Levenberg (1944). "A Method for the Solution of Certain Non-Linear Problems in Least Squares". *Quarterly of Applied Mathematics* 2: 164–168.
- [13] Donald Marquardt (1963). "An Algorithm for Least-Squares Estimation of Nonlinear Parameters". *SIAM Journal on Applied Mathematics* 11 (2): 431–441. doi:10.1137/0111030