Neural network model of the process of supercritical water oxidation of utilization of industrial effluent water

A A Tsapaev, F M Gumerov, S V Mazanov, O S Kharitonova and V V Bronskaya

1Department of Information Systems, Kazan Federal University, 35 Kremlyovskaya Street, Kazan 420008, Russian Federation
2Department of Heat Engineering, Kazan National Research Technological University, 68 Karl Marx Street, Kazan 420015, Russian Federation
3Department of Chemical Process Engineering, Kazan National Research Technological University, 68 Karl Marx Street, Kazan 420015, Russian Federation
4Department of Chemical Technology of Petroleum and Gas Processing, Kazan National Research Technological University, 68 Karl Marx Street, Kazan 420015, Russian Federation

E-mail: olga.220499@mail.ru

Abstract. In order to create a complex of control and prediction of optimal reaction conditions with a minimum value of chemical oxygen demand, a neural network model of supercritical water oxidation of industrial effluent water utilization process of hydroperoxide epoxidation of propylene at PJSC “Nizhnekamskneftekhim” was created. A full application Windows Forms, which implemented functions of loading a training sample from a file, setting the necessary training accuracy, entering a vector for obtaining results of neural network operation and graph plotting, was created.

Nowadays the share of chemical and petrochemical industries is growing rapidly as a result of which there is a question about the ecological component of these industries [1]. The main problem of any chemical production is the waste reclamation [2-4]. If inert and not very dangerous industrial waste is often disposed into the waste storage sites, toxic waste is first to be treated with thermal deactivation or physical and chemical neutralization [5-7]. One of the wastes which are under the thermal neutralization is the waste of propylene hydroperoxide epoxidation process to jointly produce styrene and propylene oxide, which is implemented at PJSC “Nizhnekamskneftekhim”. A distinctive feature of this process is using molybdenum compounds soluble in the reaction mass, which are used as a catalyst. The catalytic complex is destroyed and excreted together with the reaction products in the form of a heavy fraction during the reaction. After washing the resulting used alkaline waste makes it difficult to separate the waste by distillation or rectification due to the presence of sodium salts and residual alkali that are emulsifiers. As a result, after the thermal neutralization molybdenum complex is irretrievably lost with waste.

The possibility of using supercritical fluid technologies (SCF), in particular supercritical oxidation of waste in water medium (SCWO) is considered to solve the investigated runoff treatment and molybdenum extraction problem. The use of water in the SCF state (t > 374 ºC, P > 22.1 MPa) allows
dissolving any organic compounds containing hard-saponifiable substances, as well as complex mixtures of organic substances.

The degree of pollution of effluent water and the oxidation reaction product was determined by the values of “chemical oxygen demand” (COD) obtained from the analyzer “Expert-003-COD” photometric with the thermoreactor for 26 samples according to GOST R 52708-2007.

Oxidation efficiency X was determined by the formula:

$$X = 1 - \frac{COD_{init}}{COD_{end}}$$  \hspace{1cm} (1)

where: COD$_{init}$ is the chemical oxygen demand for the initial runoff, mg O$_2$/l; COD$_{end}$ is the value of the chemical oxygen demand for oxidized water runoff, mg O$_2$/l.

The reaction time in flow mode ($\tau$) was determined by the equation:

$$\tau = \frac{V}{Q_1 + Q_2} \times \frac{V_0}{V_r} \times 60$$  \hspace{1cm} (2)

where: V – the reactor volume (ml); V$_0$ and V$_r$ – specific volumes of the initial runoff at room temperature, atmospheric pressure and under reaction conditions (m$^3$/kg), respectively; Q$_1$– the initial water flow feed rate (ml/min); Q$_2$ – air flow rate (ml/min).

An experimental research of the process of supercritical water oxidation (T=673-873 K, P=25 MPa) of molybdenum-containing water runoff using atmospheric oxygen as an oxidizer on the installation of flow type is carried out. Besides, the growing singificance of neural network modeling is due to the growing level of complexity of modern chemical-technological systems and the need of detailed mathematical description [8-10]. In general, a neural network is a machine that is designed to simulate how the brain completes a specific task or function; a network is implemented using electronic components or created a model as software on your computer. A significant decrease in the COD value of the reaction product in comparison with the COD of the initial water flow is established. The most optimal conditions for the reaction are: T = 873 K, $\tau = 230$ s, P = 25 MPa, at which the minimum value of COD = 713 mgO$_2$/l is achieved. The obtained samples of inorganic residue contains 2.3% of mass of molybdenum.

In order to create a complex of control and prediction of optimal conditions of reaction with minimum COD value, a neural network model of supercritical water oxidation process of industrial effluent water utilization and Windows Forms application were created.

Neural network models are a universal mechanism for modeling functions and classification of objects, as well as a powerful modeling method that allows to reproduce complex dependences. Neural network model structural identification consists in selecting used activation functions, the number of network layers, the number of neurons in each layer [11]. The choice of activation function depends on the tasks for which the synthesized neural network model is supposed to be used. Typically, the activation function is a logistic function or hyperbolic tangent. These functions are applicable to a wide range of tasks. The influence of number of neurons in the hidden layer on the prediction error was studied. The algorithm of the variable metric was used as a method of minimizing the objective function for all types of architecture.

The framework implemented in C# programming language was chosen to build a neural network. The whole work will be implemented in this language. A multilayer network of direct distribution which is the most suitable topology for empirical modeling and engineering applications is used in this paper. A teacher learning method was selected from all network learning methods. The training procedure requires a set of process inputs and outputs. Weights and offsets are iteratively adjusted to minimize the target function during learning. The learning algorithm - backpropagation (reverse error propagation method) - moves the network parameters in the direction of negative gradient. The activation function is a sigmoidal with a coefficient that is equal to 1.

Initially, input and output variables are defined. The input and output data are presented in table 1 based on purposes and objectives.
Table 1. Input and output parameters of neural network.

| Input                                | Output                                    |
|--------------------------------------|-------------------------------------------|
| Excess O2, O2                        | Chemical oxygen demand, COD               |
| Reactor temperature, Tr              |                                            |

Then the architecture of artificial neural network is chosen. Figure 1 shows a diagram of the neural network.

Figure 1. Neural network topology.

The training and testing vectors are formed from a variety of input and output data. The testing pairs are presented in Table 2.

Table 2. Test sample.

| Input Vector | Oxygen % | Temperature °C | Output Vector | COD          |
|--------------|----------|----------------|---------------|--------------|
| №            |          |                |               |              |
| 1.           | 100      | 400            | 23859.0769524373 |              |
| 2.           | 200      | 450            | 14844.972149529 |              |
| 3.           | 500      | 700            | 634.573600084032 |              |
| 4.           | 400      | 500            | 5181.6962488993 |              |
| 5.           | 550      | 550            | 1735.98115300195 |              |
| 6.           | 250      | 600            | 841.247539660088 |              |
| 7.           | 300      | 700            | 643.435987003171 |              |
| 8.           | 150      | 450            | 12686.0904379294 |              |
| 9.           | 500      | 400            | 9896.77465544074 |              |
| 10.          | 350      | 500            | 6967.04138664054 |              |

The program part was written after forming a general idea of the necessary neural network structure. An artificial neural network in C# language was implemented in Visual Studio 2017 development environment using the AForge.Neuro library. A full application Windows Forms, which implemented
functions of loading a training sample from a file, setting the necessary training accuracy, entering a vector for obtaining results of neural network operation and graph plotting, was created. The application interface is presented in figures 2, 3. The keyboard input check was implemented. This check consists of that the user has the ability to enter only digits and separating characters and dots are automatically replaced with binary commas for convenience to record data in the program.

The optimal number of neurons in the hidden layer is determined by the formula [12], which is equal to 6, therefore, in Table 3, their number closest to the optimum is equal to 5.

| № | I = 1 | I = 2 | I = 3 | I = 4 | I = 5 |
|---|------|------|------|------|------|
| 1. | 0.7342381839 | 0.0332737584 | 0.0230203448 | 0.2517382682 | 0.9641764571 |
| 2. | 0.9854359308 | 0.2445463326 | 0.0853349673 | 0.3656020338 | 0.8268863934 |
| 3. | 0.9999997335 | 0.9988068689 | 0.6216481668 | 0.9546875292 | 0.0030827963 |
| 4. | 0.999521177 | 0.8369385478 | 0.7330520737 | 0.3853772179 | 0.3065432470 |
| 5. | 0.9999994492 | 0.9842959568 | 0.9673680898 | 0.4612037251 | 0.0536145444 |
| 6. | 0.994330453 | 0.9864806661 | 0.370158990 | 0.9009050882 | 0.1196887072 |
| 7. | 0.999576883 | 0.996609525 | 0.0288745219 | 0.9812102081 | 0.0112437922 |
| 8. | 0.9501488905 | 0.1995569178 | 0.033058946 | 0.4198555288 | 0.868721601 |
| 9. | 0.999856510 | 0.2175624508 | 0.9862924237 | 0.0519113631 | 0.6655821937 |
| 10. | 0.998300385 | 0.7981015899 | 0.5017879123 | 0.4403251271 | 0.3797097708 |

Table 4. Output layer.

| № | I = 1 |
|---|------|
| 1. | 0.94423005789081182 |
| 2. | 0.5930434889601377 |
| 3. | 0.0022876438284658179 |
| 4. | 0.20230384030687481 |
| 5. | 0.04188343465388196 |
| 6. | 0.020333917474099589 |
| 7. | 0.0068196203267752095 |
| 8. | 0.5930082432879894 |
| 9. | 0.39022396613515747 |
| 10. | 0.26073101136775223 |

For network training, the results were set at a relative error of 0.5%. In order to select the architecture of an artificial neural network, the deviations of the calculated values of the COD from the experimental
values were determined (Table 5). The most optimal architecture of an artificial neural network is the architecture 2-5-1, which gives the smallest deviation in COD.

**Table 5. Input and output parameters of the neural network.**

| Architecture | Error COD       |
|--------------|----------------|
| 2-1-1        | 6.117*10^{-7}  |
| 2-2-1        | 4.352*10^{-7}  |
| 2-3-1        | 4.270*10^{-7}  |
| 2-4-1        | 5.333*10^{-7}  |
| 2-5-1        | 4.115*10^{-7}  |
| 2-6-1        | 5.322*10^{-7}  |

The test adequacy of the neural network to the experiment is checked using the Fisher test. After training, the responses of the neural network to the testing effects, which were used to test the adequacy of the neural network to the experiment, were calculated.

![Figure 2. Application interface for working with neural network.](image)

![Figure 3. Application interface for displaying diagrams.](image)

Loading of input data from the neural network training file is implemented. The CSV format was selected for the files and easy writing and reading. Input vectors are written as “T, O2, COD”.
The data training and test samples were compiled during the experiments which are carried out in the laboratory.

Next step is to normalize the datasets. Neural network training is more efficient when input and output variables are normalized. Datasets have been normalized in the range [0; 1] using the formula:

\[ x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \]  

where \( x \) is the original value, \( x_n \) is the normalized value, \( x_{min} \) is the minimum value of this parameter from the set, and \( x_{max} \) is the maximum value of this parameter from the set.

COD errors were calculated to select the architecture. The results obtained during the operation of the neural network and obtained by the experiment were compared. The most optimal architecture is 2-5-1.

The adequacy check was carried out due to the determination coefficient. The value of the determination coefficient in the case of training (0.999) and testing on average (0.994) samples are close as R2 values upon checking the network for adequacy was shown and the network retraining is not observed.

The obtained results of forecasting show that artificial neural networks allow solving complex problems of property determination, in which there is no obvious dependence of the function on process parameters, and can be used to improve forecasting techniques.

An application for predicting chemical oxygen demand using artificial neural network is developed and can be used to optimize the water oxidation regimes with following implementation of process control program complex.

Acknowledgments

The work was carried out with the financial support from Russian Foundation for Basic Research in the framework of the research project № 18-29-06041.

Reference

[1] F M Gumerov 2012 American Journal of Analytical Chemistry 3 950-7
[2] K Kim, Kim K, S H Son, J Cho and Y-C Kim 2011 Supercritical Fluids 58 121
[3] A Aetov, F Gumerov, R Usmanov, Z Zaripov, I Gabitov, R Gabitov and S Mazanov 2017 Mass Spectrom. Purif. Tech 3
[4] A Vostrikov, O N Fedyaeva and A V Shishkin 2017 J. Eng Thermophysics 314
[5] A U Aetov, R A Usmanov R R Gabitov, S V Masanov and M S Kurbangaleev 2016 Vestnik of Kaz. technom. university 19 37-9
[6] B Yang, Z Shen, Z Cheng and W Ji 2017 Chemosphere 188 642
[7] D Xu, S Wang, J Zhang, X Tang, Y Guo and C Huang 2015 Chem. Eng. Res. Design 94 396
[8] Aminova G A, Manuiko G V, Bronskaya V V, Ignashina T V, Dyakonov G S, Bashkirov D V and Demidova E V 2008 Journal of Engineering Physics and Thermophysics 81 1247-51
[9] Manuiko G V, Aminova G A, Dyakonov G S, Ahmetov I G and Bronskaya V V 2015 Theoretical Foundations of Chemical Engineering 49 246-51
[10] Manuiko G V, Aminova G A, Bronskaya V V, Ignashina T V, Dyakonov G S and Bashkirov D V 2008 Theoretical Foundations of Chemical Engineering 42 336-9
[11] O S Kharitonova et al 2019 IOP Cont.Ser.: Earth Environ. Sci.315 032025
[12] Gordeev L S, Ivanov V A and Stolyarov Yu Yu 1998 Algorithms of Neural Network Modeling of Chemical-Technological Processes Software Products and Systems 25-9