Development of the intellectual complex for parallel work of steam boilers

E A Muravyova and R F Gabitov
Ufa State Petroleum Technological University, Branch in Sterlitamak, Sterlitamak, Russia

E-mail: muraveva_ea@mail.ru

Abstract. High rates of industrial and social progress require a sharp increase in heat generation on the basis of the powerful development of fuel and energy. Design of parallel operation of several boilers today is an important and necessary part of effective work in the workplace. Parallel connection of the boiler is made to: increase the maximum output and its subsequent progression, improve resiliency boilers, economical power consumption due to less loss of efficiency when operating at partial power, raising the living resource boilers, thrifty power consumption due to less loss of efficiency when operating at partial power, raising the living resource boilers, prevent local overheating and coking of tubes in the separation of heavy residues. Despite the significant advantages of using a parallel connection of the boiler there are problems such as a complex piping boiler, consumption of materials to connect the boiler, the increase in unforeseen expenditure. In the operation of the boiler unit therein may be damaged (piping, boiler elements accident water economizers) become unstable (water omission due to poor ventilation compartment exhaust emissions occur explosions and pops) that create dangerous situations fraught failure of the equipment or boiler unit as a whole, with the destruction of large material losses and loss of human life. A neural network is a system of contact between a simple processor whose operation is processing of received signals and send them to other CPUs. This processor is called a neuron. Artificial neural networks are selected for the development of process control systems due to the fact that they are fault tolerant. In a neural network information is distributed throughout the network, which means in case of failure of the neuron network behavior will be changed slightly, changing the behavior of neurons, but the network itself continues to operate successfully. Neural networks are not programmed in the usual sense of the word, they are trained. The possibility of training - one of the major advantages of neural networks over conventional algorithms. Technical training is to find the coefficients of the connections between neurons. During training, the neural network is able to identify the complex relationships between input and output data, and perform synthesis. This means that in case of successful learning network will be able to return the correct result on the basis of data that were not available in the training set.

1. Introduction
Stable operation of steam boilers in parallel mode at the electric power plant (figure 1) is ensured by accurate determination of temperature in each boiler, quantity of fuel supplied to each boiler and pressure on the common steam line going further to the raw material heating shop. The amount of fuel supplied is fixed by FLONET FN10 with FE 1a and FE 2a. Temperature downstream of boilers - temperature sensors: resistance thermometer TSRP TC-1388 with positions TE 1a and TE 2a. Common
steam line pressure - pressure sensor to position PE 1a. Due to change of valve opening degree, quantity of supplied fuel changes accordingly.

![Scheme of parallel operation of boilers.](image)

**Figure 1.** Scheme of parallel operation of boilers.

Thus, the control of the steam pressure in the common steam line during parallel operation of the boilers should ensure simultaneous control of the load of several boilers and allow to change the participation of each of them in covering the common electric load of the station.

To stabilize the mode of individual boilers not only in case of external, but also in case of internal disturbances, a control system using a neural network is used.

In case of internal disturbances, for example, when fuel supply to one of the boilers is reduced, the neural network will increase fuel supply to the boiler with low amount of fuel supplied and restore its load again to the specified value.

Possible problems of control of parallel operation of steam boilers are the following: reduction of economy, fluctuations of steam production, reduction of pressure to vacuum. Therefore, the use of a neural network to solve a given problem, rather than a standard PID regulator, will improve the quality of solving and controlling the problem, as well as the neural network has high fault tolerance and speed [1, 2, 3].

2. **Neural network development**

It is necessary to develop an artificial neural network (ANN) to control boilers in parallel operation on a common steam highway.

The development of INS was carried out in Matlab R2015b environment. The Matlab Neural Network Toolbox was used to solve the task [4].

The process of building a neural network model can be conditionally divided into 5 main stages.

The first step in building a neural network model is to carefully select the input data affecting the expected result. All information relevant to the problem under investigation must be included from the source information.

In the second step, the original data is converted taking into account the nature and type of problem displayed by her-network model, and methods of representing the formation are selected.

The third step is to design the ANN, i.e. to design its architecture (number of layers and number of neurons in each layer).

The fourth stage involves network learning, which can be conducted in a constructive or destructive manner. According to the first approach, INS training is carried out on a small network, which is gradually carried out until the required accuracy according to the test results is achieved.

In the fifth stage, the obtained INS model is tested on an independent sample of examples [5, 6].
2.1. Data collection
To solve the problem using a neural network, it is necessary to collect data for its training. A training data set is a set of observations for which input and output variable values are specified.

To create a training set for the neural network under development, we will use the data and formula obtained by experimental means:

\[ F = 100 - T_{1,2} \cdot (P_{общ})^{-1} \cdot Q_{1,2} \cdot 0.1 \] (1)

Where \( F \) – degree of opening of valve 1 and valve 2;
\( T_{1,2} \) – temperature in the boiler 1 or 2, respectively;
\( P_{общ} \) – pressure in the common steam line;
\( Q_{1,2} \) – amount of fuel supplied to the boiler 1 or 2, respectively.

Next, we will prepare data for neural network training.

2.2. Data processing and preparation
Using formula (1), create table 1 with the learning dataset. It will consist of 1000 examples.

5 values will be supplied to the neural network input:
1. Temperature is couple after a boiler 1, °C – TE 1a;
2. Temperature is couple after a boiler 2, °C – TE 2a;
3. Vapor pressure on the general steam highway, kgf/cm² – PE 1a;
4. Amount of the given fuel for heating of water in a boiler 1, t/h – FE 1a;
5. Amount of the given fuel for heating of water in a boiler 2, t/h – FE 2a.

At the output, the neural network shall calculate:
1. Degree of valve opening for fuel supply to boiler 1, %;
2. Degree of valve opening for fuel supply to boiler 2, %.

2.3. Selecting the Type and Architecture of the Neural Network
It is necessary to develop a neural network to control the parallel operation of several steam boilers.

To determine the architecture of the neural network, the possibility of using different types of networks in different classes of tasks was analyzed.

The neural network structure is selected according to the features and complexity of the task. In fact, the number of layers and the number of neurons in each layer of the neural network is limited by the resources of the computer on which the neural network under development is implemented, which will consist of two layers - hidden and output.

At the first stage of neural network construction sets of training samples are generated.

In the second step, the input data is normalized, that is, it is determined whether the measured physical value falls within the permissible limits of the sensor.

The third stage is the selection of the neural network structure. A two-layer unidirectional network will be used with a sigmoidal hidden layer neuron activation function and a linear output layer neuron activation function (created by the fitnet function). Such a network allows to solve the tasks of multidimensional approximation as precisely as possible provided data consistency and sufficient number of neurons in the hidden layer (net = fitnet(hiddenLayerSize, trainFcn)) [7].

In the fourth step, the neural network is trained on the data set. The Neural Toolbox of the Matlab package was used to build and train neural networks [8].

Temperature values from temperature sensors, quantity of supplied fuel from flow meters and pressure from pressure sensor are supplied to neural network input. Depending on each of the input values, the neural network prepares two values at the output, which in turn are supplied to the valve units, so that the degree of opening of each of them changes [9, 10, 11].

The network will be trained using the modified error backpropagating algorithm trainFcn = ‘trainbr’.

TrainBR is a network learning function that modifies weights and offsets according to the Levenberg-Marquart optimization algorithm. This minimizes the combination of error squares and weights, then determines the corrected combination, which improves the generalization capacity of the network.
Then we select the network training parameters.

### Table 1. Input variables.

| input | output |
|-------|--------|
| T1, °C | Q boiler1, t/h | % boiler 1 |
| T2, °C | Q boiler2, t/h | % boiler 2 |
| Pgen, kgf/sm² | | |
| Q boiler1, t/h | | |

| | | | | | | | |
| 791 | 449 | 30 | 29 | 29 | 22 | 56 |
| 453 | 525 | 29 | 27 | 26 | 58 | 52 |
| 532 | 544 | 29 | 25 | 30 | 54 | 45 |
| 769 | 830 | 27 | 29 | 32 | 19 | 4 |
| 338 | 337 | 30 | 29 | 27 | 68 | 69 |
| 306 | 821 | 26 | 30 | 26 | 64 | 18 |
| 379 | 312 | 31 | 30 | 30 | 63 | 70 |
| 431 | 421 | 29 | 27 | 31 | 60 | 56 |
| 608 | 548 | 32 | 29 | 29 | 45 | 51 |
| 383 | 593 | 31 | 28 | 28 | 65 | 47 |
| 792 | 402 | 30 | 28 | 27 | 26 | 63 |
| 553 | 627 | 31 | 30 | 30 | 46 | 39 |
| 551 | 314 | 26 | 31 | 31 | 35 | 63 |
| 679 | 612 | 27 | 31 | 28 | 20 | 36 |
| 596 | 823 | 26 | 28 | 32 | 36 | 1 |
| 372 | 571 | 31 | 25 | 32 | 70 | 41 |
| 625 | 265 | 30 | 25 | 31 | 48 | 73 |
| 670 | 648 | 25 | 30 | 32 | 20 | 18 |
| 604 | 634 | 25 | 26 | 26 | 37 | 34 |
| 564 | 470 | 31 | 29 | 28 | 47 | 57 |
| 278 | 358 | 25 | 31 | 27 | 66 | 62 |
| 697 | 764 | 27 | 32 | 28 | 17 | 19 |

**Figure 2.** Structural diagram of ANN.

We set the maximum number of epochs of learning, which determines the number of epochs (time interval) after which learning will be terminated:

```python
net.trainParam.epochs = 1000.
```

We will choose the number of epochs between shows equal to five:

```python
net.trainParam.show = 5.
```

We specify the completion criterion - the deviation value at which the training will be considered completed:
Next, we divide the data into a Training set, a Validation set, which is used to evaluate the generalizing properties of the network and stop learning when generalization stops improving and a Testing set that does not affect training, but serves to test on data that was not used in network training:

\[
\text{net.divideParam.trainRatio} = \frac{60}{100}; \\
\text{net.divideParam.valRatio} = \frac{35}{100}; \\
\text{net.divideParam.testRatio} = \frac{5}{100}.
\]

### 3. Building and Learning a Neural Network at Matlab

Next we implement and train the neural network in Matlab. To do this, enter the uiopen command. The neural network input is called input and the output is called output. The downloaded data is displayed in «Workspace» [11, 13, 14].

Input data - basic - is entered in the form of matrix with number of rows - 48 and columns - 5, on which output data will be selected. Another array of data that will be output by the neural network consists of 48 rows and 2 columns.

Using the nnstart command, you will enter the Neural Network Learning tab, where you will implement the approximation and regression task. At this stage, network input data and network output data are selected. Since matrices are specified in columns, under «Samples are», we select «Matrix rows».

Then the number of hidden layers of neurons - 50 is selected. As a result the neural network will be presented by four blocks: entrance, hidden layer of neurons, output layer of neurons, exit (figure 3). The «input» unit implements the process of reading the input data, in our case it is the steam temperature after boilers No. 1 and No. 1, the pressure in the common steam line and the fuel consumption supplied to each boiler, and transfers them to the hidden layer of the developed neural network. The adder "multiplies each input by the weight and sums the weighted inputs. Then the value passes through the function of activation of the corresponding layer and the output is calculated: regulation of the degree of valve opening before boilers No. 1 and No. 2.

This structure allows monitoring of training progress, as well as calculation of static results and display of training quality estimates [12, 15, 16].

![Figure 3. Neural Network Block Structure.](image)

The next stage of neural network formation is selection of algorithm of developed INS. In this work, the "Bayesian regulation" algorithm was chosen, as it works more accurately than the others.

Neural network training lasts depending on the number of said hidden layers, respectively, the more layers, the longer the training will take. During training, iterations are run, that is, the number of passes, each of which passes the specified algorithm - straight and reverse passes. The epoch, in turn, is the number of times an algorithm reads the entire dataset, the matrix. Thus, each time the algorithm counted all samples in the dataset, the epoch is completed [8, 9, 11].
In the network learning process window, by clicking on the Performance button, you can see the network learning schedule showing the learning error behavior shown in figure 4.

![figure 4. Mean quadratic error.](image)

For 1000 eras the value of a mean square mistake $3,2371 \cdot 10^{-11}$ is reached. This means that the values of the value under study lie in close proximity to each other, which means that the error of the values will be minimal. The learning feature uses early stopping learning as an anti-retraining tool. MSE is a function of network performance. It shows the performance according to the mean quadratic error [10, 12].

Figure 5 shows the Training State graphs. The "val fail" graph shows the error change on the control set. The val fail parameter shows the number of iterations where the error was less than the specified one. There were no such iterations, indicating the exact operation of the neural network.

![figure 5. Network Training Schedules.](image)
The graph "ssX" shows the average sum of quadratic parameters per 1000 epochs, which introduce errors into the final result of operation of the neural network. The value of the "ssX" function is 9.2143, which is due to the minimum effect of errors on the total training result of the neural network.

The graph "gamk" shows the average number of parameters per 1000 eras that affected the accuracy of the result achieved. The value of the function "gamk" takes the value 81.9949, which shows that the influence of external factors, such as errors, occurred frequently.

The "mu" graph reflects the change in the training parameter [pi] of the Bayesian regularization method, and the higher the given value [pi], the more accurate the network training will be. A value of "mu" of 50 means that the mathematical expectation has reached the upper operable limit at 1000 epoch.

The "gradient" graph shows the function gradient change across the network weights at each iteration. At each subsequent iteration, the error approaches zero. A value of grade = 0.00034069 means that the error is virtually minimal.

As a result of training, a neural network is obtained, which without reconfiguration of link weights generates output signals when any set of input signals from the training set are supplied to the network input [8, 9].

4. Check of neural network
To test the neural network, use the sim command (net, [x1; x2; x3; x4; x5]) to input 5 values. The check inputs shown in Table 2 are randomly selected according to the rules and upper/lower measurement boundaries. Based on them, the neural network calculates the output data of ANN [15].

Using text commands the correct training of ANN for control of parallel operation of steam boilers is checked. As a result of testing, the truth of the output values is (55.2311; 58.1423). Proximity of the obtained values to the given result (55; 58) indicates the applicability of the network. It follows that the parameters of neural network data are selected correctly [16].

Table 2. Verification data.

| Input                              | Output                        |
|------------------------------------|-------------------------------|
| Temperature is couple later copper 1, °C | Temperature is couple after a copper 2, °C | Steam pressure on the common steam line, kgf/cm² | Amount of fuel supplied to boiler 1, t/h | Amount of fuel supplied to boiler 2, t/h | Extent of opening of the valve, % | Extent of opening of the valve, % |
| 430                                | 436                           | 26                           | 27                           | 25                           | 55.2311                           | 58.1423                           |

5. Conclusion
In the course of solving the problem of controlling parallel operation of steam boilers, the expediency of developing an automatic control system based on a neural network with parameters of steam boilers during their parallel operation is justified and shown, taking into account the mutual influence of parameters of the object.

A model for controlling the steam boiler system is presented, taking into account the relationships between the process parameters. During neural network development, the most optimal number of hidden layers is selected, the selected training algorithm is effective at the specified task and at the specified conditions and parameters.

Stable operation and low RMS error result in high-quality use of neural network to control boilers in real time.
References

[1] Muravyova E and Popkov V 2019 Development and Research of a Dynamic Flow Laboratory Bench Model 7th Scientific Conference on Information Technologies for Intelligent Decision Making Support (Ufa: (ITIDS)) pp 117-22

[2] Yakhyaeva G 2006 Fuzzy Sets and Neural Networks (Moscow: Internet University of Information Technologies; Binom) p 316

[3] Popov A, Sayarkin K and Zhylenkova A 2018 The scalable spiking neural network automatic generation in MATLAB focused on the hardware implementation (Moscow: IEEE) pp 285-287

[4] Rashid T 2016 Make Your Own Neural Network (North Charleston: publisher: «CreateSpace») p 222

[5] Muravyova E and Gabitov R 2018 Economic Features to Optimize the Catalyst Calcinations Process International Scientific Multi-Conference on Industrial Engineering and Modern Technologies (Vladivostok: IEEE) p 178

[6] Muravyova E, Sharipov M and Gabitov R 2018 SCADA-System Based on Multidimensional Precise Logic Controller for the Control of a Cement International Scientific Multi-Conference on Industrial Engineering and Modern Technologies (Vladivostok: IEEE) p 260

[7] Kriesel D 2014 Brief Introduction to Neural Networks (Peru: «Autoedicin») p 244

[8] Samarasinghe S 2016 Neural Networks for Applied Sciences and Engineering (Boca Raton: «CRC Press») p 570

[9] Evmenov V 2009 Intelligent Control Systems (Moscow, Russia: book house "Librokom") p 290

[10] Muravyova E and Almakaev I 2019 Electrical Heating Reactor Control System Using Neural Network and the Fuzzy Controller International Multi-Conference on Industrial Engineering and Modern Technologies (Vladivostok: (FarEastCon)) pp 250-6

[11] Muravyova E and Sharipov M 2018 Intelligent Control System for Process Parameters Based on a Neural Network Actual problems of electronic instrument engineering (Novosibirsk: APEIE) pp 256-60

[12] Haikin S 2006 Neural Networks: a Complete Course (Moscow: Publishing house «Williams») p 1104

[13] Barsky A 2004 Neural Networks: Recognition, Management, Decision-Making (Moscow, Russia: «Finance and statistics») p 176

[14] Andrzej Cichocki and Seungjin Choi 1998 Cascade Neural Networks for Multichannel Blind Deconvolution (Stevenage: Electronics Letters vol. 34) pp 1186-7

[15] Sherfey Jason S, Soplata Austin E, Ardid Salva, Roberts Erik A, Stanley David, Pittman-Polletta Benjamin R and Kopell Nancy J DynaSim 2018 A Matlab Toolbox for Neural Modeling and Simulation (Oslo: Published in Frontiers in Neuroinformatics) pp 1183-92

[16] Muravyova E and Sharipov M 2018 Method of Optimal Parameters Control in Three-Phase Separator using Fuzzy Controller International Conference on Actual Issues of Mechanical Engineering (Novosibirsk: AIR) pp 418-23