ISSN 2786-7102 (Online), ISSN 2786-7633 (Print)
https://doi.org/10.15407/srenergy2025.01.074

UDC 531.7:62-2:629.7 (043.3)

Janusz Kacprzyk[1], Dr. Sci. (Engin.), Professor, https://orcid.org/0000-0003-4187-5877
Artur Zaporozhets[2], Dr. Sci. (Engin.), Senior Researcher, https://orcid.org/0000-0002-0704-4116
Denys Kataiev[2]*, https://orcid.org/0000-0002-2383-3123
[1]Systems Research Institute, Polish Academy of Sciences, 6, Newelska St., Warsaw, 01-447, Poland;
[2]General Energy Institute of NAS of Ukraine, 172, Antonovycha St., Kyiv, 03150, Ukraine
*Corresponding author: kataev_d.a@ukr.net

DEVELOPMENT OF AN ARTIFICIAL NEURAL NETWORK FOR INFORMATION AND MEASUREMENT SYSTEM FOR CONTROLLING THE GEOMETRIC DIMENSIONS OF POWER EQUIPMENT

Abstract. *The paper deals with the development of an artificial neural network for compensating for nonkinematic errors of an information and measurement system (IMS) based on a coordinate measuring arm (CMA). After compensating for kinematic errors using a mathematical model, the proposed back-propagation neural network corrects non-kinematic errors arising from thermal deformations, noise, and element deformation inaccuracies. Experimental studies conducted on synthetic data demonstrated a significant reduction in the mean square error (MSE) of the coordinates of the measured points and a decrease in measurement uncertainty. The model exhibited high accuracy and stability, which confirms its effectiveness for controlling the geometric parameters of energy equipment.*

Keywords: artificial neural network, error back propagation, error compensation, information and measurement system, coordinate measuring arm, geometric parameters, modeling.

1. Introduction

Information and measurement systems (IMS) based on the coordinate measuring arm (CMA) allow for complex three-dimensional measurements with high accuracy, which is especially important in critical industries such as the production and maintenance of turbines, generators, pumps and other components for nuclear power plants (NPPs), thermal power plants (TPPs) and hydroelectric power plants (HPPs). IMS are used at all stages of the equipment life cycle, providing quality control, part reproduction, and the development of new solutions.

Sources of errors in information and measurement systems based on a coordinate measuring arm are usually divided into kinematic errors (such as link length errors, link twist errors, and initial zero position errors) and nonkinematic errors (such as link deformation errors, thermal errors, and encoder rotary shaft motion errors) [1−3].

Most of the research in this area has focused on compensating for kinematic errors by calibrating parametric models and ignored the compensation of nonkinetic errors. There are various factors that can affect the motion uncertainty of a CGM, such as structural parameter error, deformation caused by the measuring force and gravity, rotational error of the joints, and thermal deformation. Using classical mathematical methods, it is difficult to model and compensate for all these factors [2−4].

Therefore, we propose an approach in which, after compensating for kinematic errors by mathematical methods using the classical Denavit-Hartenberg model, a back-propagation neural network is used to compensate for nonkinematic errors.

2. Methods and materials

The Back-Propagation Neural Network (BPNN) (Fig. 1) is one of the most widely used artificial neural networks with a powerful ability of nonlinear mapping and self-learning. Thanks to this ability, BPNN can predict the errors of the Information and Measurement System caused by various factors, which makes it possible to compensate for complex errors and reduce the uncertainty of the Information and Measurement System measurements [5−8].

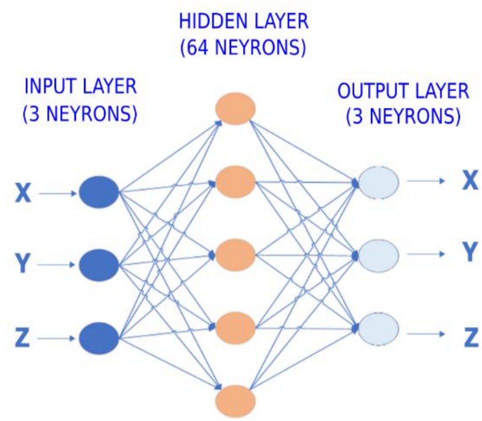

Figure 1. Back-Propagation Neural Network

The neural network is implemented as a BackpropagationNN class based on the PyTorch library. The neural network consists of three main components: an input layer, one hidden layer, and an output layer.
The input layer takes as input the coordinates of a point in three-dimensional space:

$$x = [x_m, y_m, z_m], \tag{1}$$

where: x_m, y_m, z_m − coordinates of the point obtained by the measuring system. The dimension of the input layer corresponds to the number of coordinates.

The hidden layer performs a linear transformation of the input data using a nonlinear activation function $(\mathrm{Re}\,LU)$:

$$h = \mathrm{Re}\,LU\left(W_1 x + b_1\right), \tag{2}$$

where: $W_1 \in R^{64\times3}$ − hidden layer weight matrix, $b_1 \in R^{64}$ — displacement vector, $\mathrm{Re}\,LU(z) = \max(0, z)$ − $\mathrm{Re}\,LU$ activation function that adds nonlinearity to the model. The output layer performs a linear transformation of the hidden layer output to obtain the predicted coordinates:

$$\hat{y} = W_2 h + b_2, \tag{3}$$

where: $W_2 \in R^{3\times 64}$ − matrix of weights of the output layer, $b_2 \in R^3$ − displacement vector. As an output, the network returns the corrected coordinates of the point:

$$\hat{y} = \left[\hat{x}, \hat{y}, \hat{z}\right]. \tag{4}$$

The Mean Square Error (MSE) (MSE) function is used to evaluate the accuracy of a model [9−12]. It calculates the average difference between the true coordinate (y) and the predicted coordinates (ŷ):

$$(MSE) = \frac{1}{N}\sum_{i=1}^{N} \| \hat{y}_i - y_i \|^2, \tag{5}$$

where: $y_i = \left[x, y, z\right]$ − true coordinates of the point, $\hat{y}_i = \left[\hat{x}_i, \hat{y}_i, \hat{z}_i\right]$ − predicted coordinates of the point, N − number of points in the data set.

The loss function (MSE) is minimized during training to maximize model accuracy, and the error backpropagation algorithm calculates gradients of the loss function to update the weights and biases in the network.

Calculating the error of the output layer

$$\delta_{out} = \hat{y} - y, \tag{6}$$

where: $\delta_{out} \in R^3$ − error in the original layer.

Gradients of weights and offsets of the original layer

$$\frac{\partial MSE}{\partial W_2} = \delta_{out} h^T, \ \frac{\partial MSE}{\partial b_2} = \delta_{out}. \tag{7}$$

Calculating the hidden layer error

$$\delta_{hidden} = \left(W_2^T \delta_{out}\right) \cdot \mathrm{Re}\,LU'(h), \tag{8}$$

where: $\mathrm{Re}\,LU'(z) = \begin{cases} 1, z > 0, \\ 0, z \le 0 \end{cases}.$

Gradients of weights and offsets of the hidden layer

$$\frac{\partial MSE}{\partial W_1} = \delta_{hidden} x^T, \ \frac{\partial MSE}{\partial b_1} = \delta_{hidden}. \tag{9}$$

Weights and offsets are updated using the Adam optimization algorithm:

$$W_i \leftarrow W_i - \eta \cdot \frac{\partial MSE}{\partial W_1}, \tag{10}$$

where: η − learning speed.

Adam additionally takes into account the moments of first- and second-order gradients for faster convergence. After training the model, the uncertainty of the corrected coordinates is estimated. For this purpose, confidence intervals are used:

$$Cl = t \cdot \frac{\sigma}{\sqrt{n}}, \tag{11}$$

where: t − this is the quantilic of the Student distribution, σ − the standard deviation of the predicted coordinates, n − the number of points.

The standard deviation of the projected coordinates was calculated by the formula:

$$\delta = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{\left(x_i + \hat{x}_i\right)^2 + \left(y_i + \hat{y}_i\right)^2 + \left(z_i + \hat{z}_i\right)^2}\right)^2} \tag{12}$$

where x_i, y_i, z_i, are true coordinates, $\hat{x}_i, \hat{y}_i, \hat{z}_i$ are the predicted coordinates and N is the number of points in the data set.

To build and train a neural network, we have conducted an experimental simulation of the process of calibration of the Information and Measurement System [13−15]. In the context of this work, uncertainty is a characteristic of a random component after amending systematic influences that are adjusted by a neural network. Systematic errors caused by influences, such as thermal deformations, noise, elements of deformation of the elements, are offset by teaching the network on synthetic data and using appropriate algorithms. However, the residual errors remaining after correction consist of unchanged components of a systematic error and a random component. Thus, uncertainty after correction determines the distribution of these residual errors. This means that the trust intervals that are evaluated after processing by the neural network characterize the influence of both adjusted and non -co -entered components of errors, which causes their complex nature.

A synthetic dataset consisting of three-dimensional coordinates was used for the experiment. The data was generated around a sphere of 30mm diameter. The data was generated using a normal distribution. To simulate destabilizing factors that correspond to real distortions that may occur during measurement, noise was added to the coordinates according to the Gaussian distribution (Fig. 2) [16].

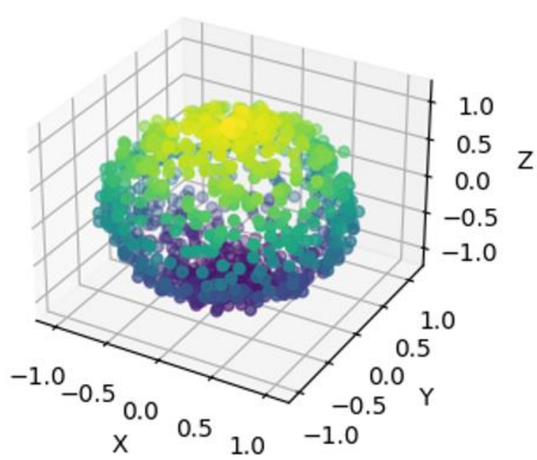

Figure 2. Adding noise according to the Gaussian distribution

The generated data were divided into training and test sets in the ratio of 80 % to 20 % using the stratified k-fold cross-validation method (Fig. 3) [17−19].

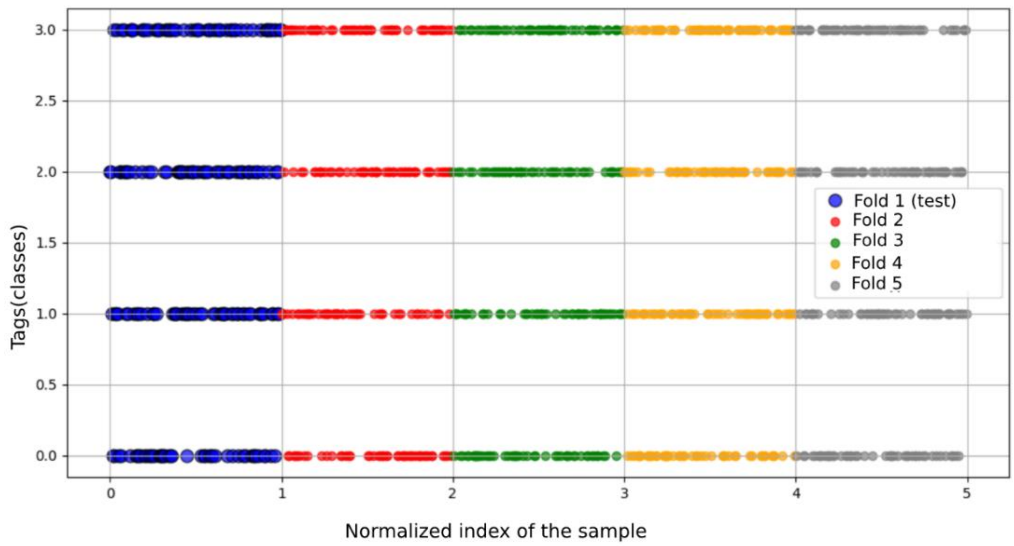

Figure 3. Stratified k-fold cross-validation

The training set was used to train the model, and the test set was used to evaluate its accuracy. Key generation parameters include: number of samples = 1000; sphere diameter = 30 mm.

An experimental study was conducted to determine the impact of different activation functions on the model's accuracy. The purpose of the experiment is to compare the effect of activation functions on model accuracy by conducting 100 training iterations for each activation function. The results were evaluated based on the average test losses obtained during each iteration (Fig. 4).

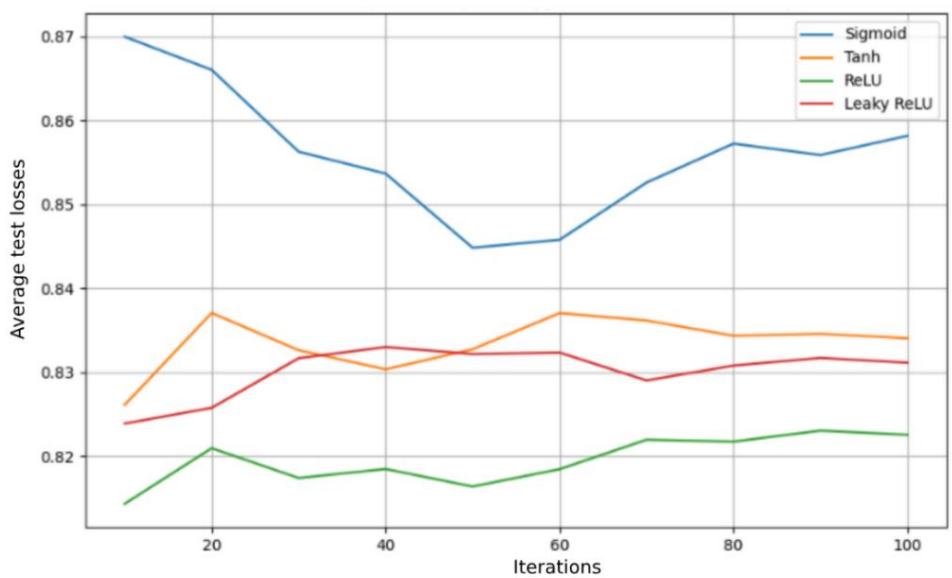

Figure 4. Average test losses for different activation functions

The ReLU activation function showed the best performance, achieving the lowest average loss over 100 training epochs. We also conducted a series of experiments to substantiate the selected hyperparameters [20].

To conduct the experiment, we used a combination of typical key hyperparameters that affect the performance of a neural network (Table 1).

Table 1. Typical key hyperparameters

Hyperparameters	***Typical values***			
Number of neurons	32	64	128	256
Speed of learning	0.001	0.01	0.1	
Number of epochs	50	100	200	
Size of mini-packs	16	32	64	128
Regularization	0.0001	0.001	0.01	

Thus, 432 possible combinations were formed to experimentally substantiate the choice of hyperparameters. The neural network was trained on the coordinates of the sphere surface using 432 typical combinations of hyperparameters, noisy data were processed by the neural network, and the predicted coordinates of 10 test points were obtained, followed by a comparison of the predicted coordinates with the reference coordinates and confidence intervals [21−23].

We also selected the appropriate metrics to determine how changes in hyperparameters affect the measurement results. Train Loss - losses on training data. Validation Loss - losses on validation data. Mean Square Error (MSE) of the predicted coordinates of the control points. Reduction of uncertainty before and after the coordinates are processed by the neural network. Correction rate, which indicates the percentage of model predictions that required correction after processing.

Models with zero correction rate were chosen as stable. This means that all the predicted coordinates after processing by the neural network fell within the confidence interval, and no corrections were required [24].

Among the models with zero correction rate, uncertainty reduction was used as the main criterion for selecting the most accurate models. Based on the comparative analysis, model No. 127 was selected (Table 2), which has a zero correction rate, which means that the predictions are stable (all predicted coordinates fell within the confidence interval) and which showed the greatest reduction in uncertainty, indicating the ability to provide the highest accuracy [25].

Table 2. Parameters of model No. 127

Model №	127
Number of neurons	64
Learning speed	0,001
Number of learning epochs	100
The size of mini-batches	64
Regularization	0,0001
Train Loss	0,027
Validation Loss	0,026
Uncertainty X before processing (mm)	8,91
Uncertainty X after processing (mm)	8,38
Uncertainty Y before processing (mm)	8,76
Uncertainty Y after processing (mm)	8,02
Uncertainty Z before processing (mm)	8,2
Uncertainty Z after processing (mm)	7,95
MSE	0,009
Frequency of correction	0

Thus, it has been experimentally determined that the best model performance in terms of stability and reduction of uncertainty is provided by the following hyperparameters: number of neurons: 64; learning rate: 0.001; number of epochs: 100; mini-packet size: 64; regularization: 0.001.

A synthetic data set was generated for the training of the neural network, which simulates the process of measuring on the surface of the sphere with a diameter of 30 mm (coordinate range: from -15 mm to 15 mm). To model random errors to the coordinates, a noise with a normal distribution was added, the average value of which was -0.1 mm, and the average square deviation was 0.1 mm. Thus, the ratio between the average quadratic deviation of noise and the maximum coordinate is approximately 0.67 %. This ensures the realism of the model, allowing to evaluate the effect of small random errors on the accuracy of projected coordinates. For each coordinate of the measured points, confidence intervals were calculated with a confidence level of 95 % [26]. The real and predicted coordinates were compared according to the criterion of finding the predicted coordinates within the confidence interval (Fig. 5).

To prevent overtraining, the model used an early stopping technique with the parameter `patience' = 20. This means that training was stopped if there was no improvement in the loss function on the validation data within 20 epochs. The model was trained on 80 % of the total generated data, while 20 % of the data was used for validation.

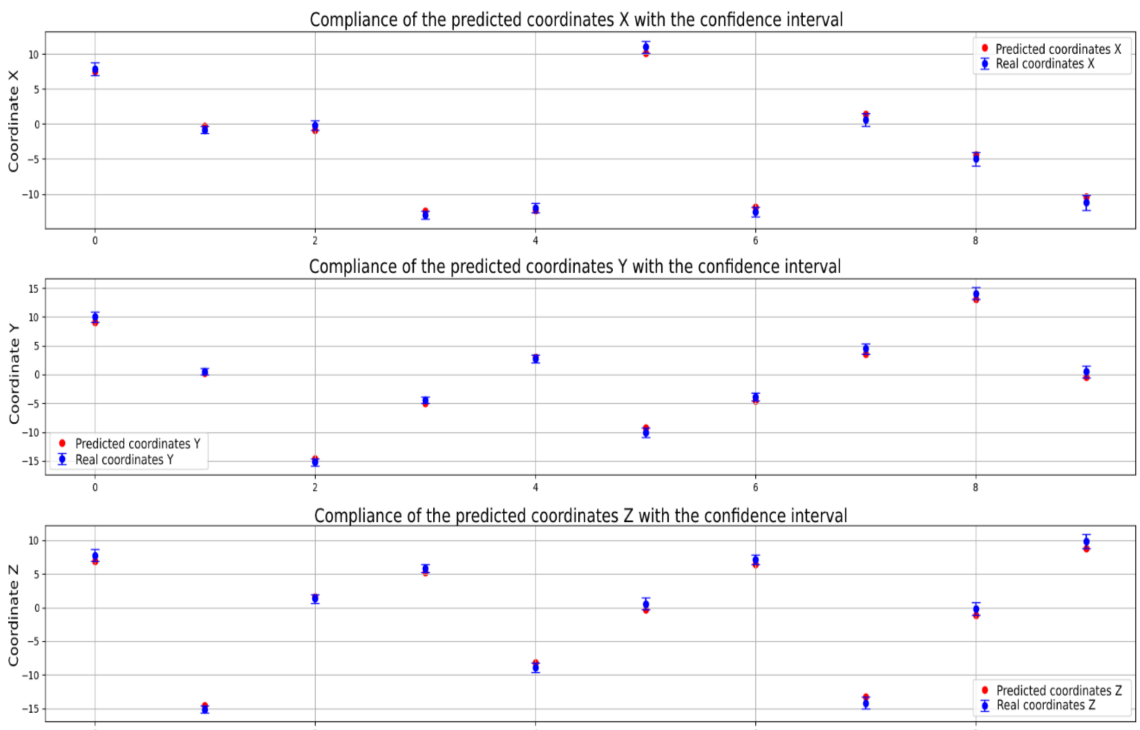

Figure 5. Correspondence of the predicted coordinates to the confidence interval

The criterion for stopping the training was the stabilization of the loss function on the validation data within 20 epochs, which allowed to avoid overtraining.

3. Results

After training, the model was evaluated on a test dataset that was not used for training and validation (Table 3).

Table 3. Measurement result

Point	*X before processing (mm)*	*X after processing (mm)*	*Y before processing (mm)*	*Y after processing (mm)*	*Z before processing (mm)*	*Z after processing (mm)*	*Distance to the sphere surface before processing*	*Distance to the sphere surface after processing*
1	4,09	5,11	-13,32	-13,98	-1,02	0,15	1,03	0,12
2	1,38	1,36	0,56	1,31	13,96	14,7	0,96	0,18
3	-4,4	-3,36	-12,95	-14	3,59	4,63	0,86	-0,12
4	5,5	5,93	-9,03	-8,89	-10,19	-10,62	0,32	-0,07
5	5,68	6,03	-5,42	-7,15	9,92	11,66	2,35	0,05
6	-2,3	-1,99	9,55	9,24	-12,09	-11,98	-0,59	-0,26
7	0,98	0,83	-13,64	-14,09	-5,93	-5,49	0,09	-0,15
8	11,68	11,43	7,66	8,97	3,53	2,21	0,59	0,30
9	-5,94	-6,31	7,24	6,86	-11,4	-11,69	0,25	0,05
10	-5,26	-5,05	-7,6	-8,14	10,51	11,46	1,00	0,06

The distance of the point from the center of the reference sphere (r) is determined:

$$r = \sqrt{x^2 + y^2 + z^2}\ , \tag{13}$$

where: x, y, z – point coordinates, r – distance from the point to the center.

If the diameter of the sphere $d = 30\text{мм}$, then:

$$R = \frac{d}{2} = 15\text{мм}. \tag{14}$$

Determine how far the point deviates from the surface of the reference sphere:

$$\Delta r = |r - R|, \tag{15}$$

where: r – calculated distance of the point, $R = 15\text{мм}$.

If $\Delta r_{after} < \Delta r_{before}$, then the point has approached the surface of the sphere (+), otherwise it has moved away (-) (Table 4).

Table 4. The result of the calculations

Point	r_{before} *(mm)*	r_{after} *(mm*	*Deviations before processing (mm)*	*Deviations after processing (mm)*	*Result*
1	13.97	14.89	1.03	0.11	+
2	14.04	14.82	0.96	0.18	+
3	14.14	15.12	0.86	-0.12	+
4	14.68	15.07	0.32	-0.07	+
5	12.65	14.95	2.35	0.05	+
6	15.58	15.26	-0.58	-0.26	+
7	14.91	15.14	0.09	-0.14	-
8	14.41	14.70	0.59	0.30	+
9	14.75	14.95	0.25	0.05	+
10	14.00	14.94	1.00	0.06	+

After processing with the neural network, the coordinates for most of the predicted points (9 out of 10) became closer to the reference surface of the sphere (Fig. 6).

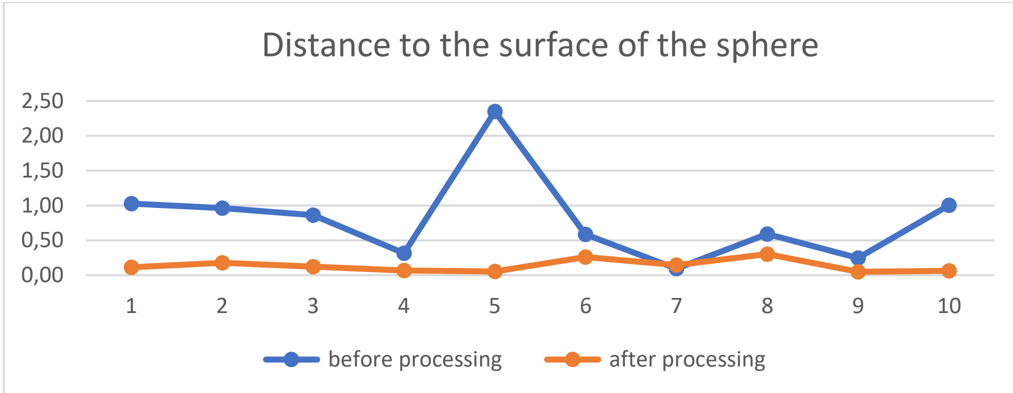

Figure 6. Distance to the surface of the sphere

The average deviation of the coordinates after neural network processing decreased from 0.8 mm to 0.14 mm.

Thus, on the basis of experimental studies, the effectiveness of using a neural network to control the geometric dimensions of power equipment was confirmed.

The proposed neural network, integrated into the TDF software system, allows correcting non-kinematic measurement errors in real time.

4. Conclusions

Experimental studies were conducted to evaluate the effectiveness of the proposed information and measurement system (IMS) based on a coordinate measuring arm (CMA) and an artificial neural network.

The architecture of the neural network for compensation of measurement errors is proposed, which allows to improve the results of controlling the geometric parameters of equipment.

The influence of various neural network training parameters on measurement accuracy and data processing speed is investigated.

Comparison of the results before and after neural network processing showed a significant increase in the accuracy of predicting the measured parameters.

Author Contributions. Methodology and ideology of the research, data analysis, manuscript preparation – Artur Zaporozhets; methodology and ideology of the research – Janusz Kacprzyk; data collection and analysis, preparation of the initial manuscript – Denys Kataiev.

Funding. The paper was supported by the departmental scientific program of the NAS of Ukraine "Development of methods and means of monitoring the state of the environment of energy facilities based on wireless sensor networks" (0123U100127).

References

1. Xuefeng, Y., Ci, S., Jin, Y., Lu, Y., & Hongwei, Z. (2016). Development and error compensation of the high precision turntable. *Chinese Journal of Scientific Instrument*, *37*(5), 961–967.
2. Zhaorui, Z., Xu, Z., & Zelong, Z. (2015). Hand-eye calibration method fusing rotational and translational constraint information. *Chinese Journal of Scientific Instrument*, *36*(11), 2443–2450.
3. Cheng, Y., Chen, X., & Wang, H. (2016). Measurement uncertainty estimation and analysis based on accuracy theory. *Journal of Electronic Measurement and Instrumentation*, *30*(8), 1175–1182.

4. Cheng, Y., Wang, Z., Chen, X., Li, Y., Li, H., Li, H., & Wang, H. (2016). Evaluation and Optimization of Task-oriented Measurement Uncertainty for Coordinate Measuring Machines Based on Geometrical Product Specifications. *Applied Sciences*, *9*(1), 6–15. https://doi.org/10.3390/app9010006
5. Tung, T. T., Tinh, N. V., Thao, D. T. P., & Minh, T. V. (2023). Development of a prototype 6 degree of freedom robot arm. *Results in Engineering*, 18. https://doi.org/10.1016/j.rineng.2023.101049
6. Wen, Z. M., Wang, Y. J., & Di, N. (2014). On-orbit hand-eye calibration using cooperative target. *Chinese Journal of Scientific Instrument*, *35*(5), 1005–1012.
7. Zhuang, J., Wang, B., & Xiang, M. (2015). Application of MEMS IMU random error modeling in SAR motion compensation. *Foreign Electronic Measurement Technology*, *34*(10), 88–94.
8. Rongsheng, L., Wanhong, L., & Dabao, L. (2014). Angular error compensation for laser tracker. *Optics and Precision Engineering*, *22*(9), 2299–2305. http://doi.org/10.3788/OPE.20142209.2299
9. Wang, Y., Xue, Z., Huang, Y., & Wang, X. (2016, January 26). Study on self-calibration angle encoder using simulation method. *Seventh International Symposium on Precision Mechanical Measurements*, 9903. https://doi.org/10.1117/12.2217638
10. Zaporozhets, A., & Sverdlova, A. (2021). Predicting anomaly conditions of energy equipment using neural networks. E3S Web of Conferences, 280, 10. https://doi.org/10.1051/e3sconf/202128009005
11. Xiong, H., Xu, Y., Zong, Z., & Gao, Q. (2012). The multi-joint measuring arm's error compensation method based on the pose configurations. *Applied Mechanics and Materials*, 117-119, 751–755. https://doi.org/10.4028/www.scientific.net/amm.117-119.751
12. Lopez-Franco, C., Hernández-Barragán, J., Alanis, A. Y., Arana-Daniel, N., & López-Franco, M. (2018). Inverse kinematics of mobile manipulators based on differential evolution. *International Journal of Advanced Robotic Systems*, *15*(1), 27–38. https://doi.org/10.1177/1729881417752738
13. Bing, H., Linxian, C., Chusheng, L., & Yonggang, H. (2014). Complex differential evolution algorithm for inverse kinematics of spatial 6R robot manipulators. *Journal of Mechanical Engineering*, *50*(15), 45–52. http://www.cjmenet.com.cn/CN/Y2014/V50/I15/45
14. Babak, S., Babak, V., Zaporozhets, A., & Sverdlova, A. (2019). Method of Statistical Spline Functions for Solving Problems of Data Approximation and Prediction of Objects State. *SSRN Electronic Journal*. http://doi.org/10.2139/ssrn.3734104
15. Asmai, E., Hennebelle, F., Coorevits, T., Rénald, V., & Fontaine, J.-F. (2018). Rapid and robust on-site evaluation of articulated arm coordinate measuring machine performance. *Measurement Science and Technology*, *29*(11), 1–14. https://doi.org/10.1088/1361-6501/aade10
16. Cheng, W., Yu, L., & Fei, Y. (2011). Study on parameter identification algorithms for articulated arm coordinate measuring machine. *University Science Technology China*, 41, 45–49.
17. Gao, G., Zhang, H., Wu, X., & Guo, Y. (2016). Structural Parameter Identification of Articulated Arm Coordinate Measuring Machines. *Mathematical Problems in Engineering*, 3, 1–10. https://doi.org/10.1155/2016/4063046
18. Huang, J., Cao, Y., Xiong, C., & Zhang, H.-T. (2018). An echo state Gaussian process-based nonlinear model predictive control for pneumatic muscle actuators. *IEEE Transactions on Automatic Control*, *16*(3), 1071–1084. https://doi.org/10.1109/TASE.2018.2867939
19. Gao, G., Sun, G., Na, J., Guo, Y., & Wu, X. (2018). Structural parameter identification for 6 DOF industrial robots. *Mechanical Systems and Signal Processing*, 113, 145–155. https://doi.org/10.1016/j.ymssp.2017.08.011
20. Luo, Z., Liu, H., Li, D., & Tian, K. (2018). Analysis and compensation of equivalent diameter error of articulated arm coordinate measuring machine. *Measurement and Control*, *51*(5), 16–26. https://doi.org/10.1177/0020294018755324
21. Babak, S., Babak, V., Myslovych, A., Zaporozhets, A., & Zvaritch, V. (2020). Diagnostic Systems For Energy Equipments. *Studies in Systems, Decision and Control*, 281, 71–90. https://doi.org/10.1007/978-3-030-44443-3_3
22. Huang, J., Ri, S., Fukuda, T., & Wang, Y. (2019). A disturbance observer based sliding mode control for a class of underactuated robotic system with mismatched uncertainties. *IEEE Transactions on Automatic Control*, *64*(6), 2480–2487. https://doi.org/10.1109/tac.2018.2868026
23. Gao, G., Zhang, H., San, H., Wu, X., & Wang, W. (2019). Kinematic calibration for industrial robots using articulated arm coordinate machines. *International Journal of Modelling, Identification and Control*, *31*(1), 16–26. https://doi.org/10.1504/ijmic.2019.096816
24. Zaporozhets, A., & Kataiev, D. (2024). Method of compensating for instrumental uncertainty in measurements using a coordinate measuring arm. *System Research in Energy*, *1*(76), 45–53. https://doi.org/10.15407/srenergy2024.01.045

25. Wan, L., Su, Y., Zhang, H., Tang, Y., & Shi, B. (2019). Global fast terminal sliding mode control based on radial basis function neural network for course keeping of unmanned surface vehicle. *International Journal of Advanced Robotic Systems*, *16*(2), 1–12. https://doi.org/10.1177/1729881419829961
26. Park, B.-J., Pedrycz, W., & Oh, S.-K. (2010). Polynomial-based radial basis function neural networks (P-RBF NNs) and their application to pattern classification. *Applied Intelligence*, 32, 27–46. https://doi.org/10.1007/s10489-008-0133-z

РОЗРОБКА ШТУЧНОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНОЇ СИСТЕМИ КОНТРОЛЮ ГЕОМЕТРИЧНИХ РОЗМІРІВ ЕНЕРГЕТИЧНОГО ОБЛАДНАННЯ

Януш Кацпжик[1], д-р техн. наук, професор, https://orcid.org/0000-0003-4187-5877
Артур Запорожець[2], д-р техн. наук, ст. досл., https://orcid.org/0000-0002-0704-4116
Денис Катаєв[2]*, https://orcid.org/0000-0002-2383-3123
[1]Інститут системних досліджень, Польська академія наук, вул. Невельська, 6, Варшава, 01-447, Польща;
[2]Інститут загальної енергетики НАН України, вул. Антоновича, 172, Київ, 03150, Україна
*Автор-кореспондент: kataev_d.a@ukr.net

Анотація. *У статті розглядається розробка штучної нейронної мережі для компенсації некінематичних похибок інформаційно-вимірювальної системи (ІВС) на базі координатно-вимірювальної руки (КВР). Після компенсації кінематичних похибок за допомогою математичної моделі, запропонована нейронна мережа зі зворотним поширенням помилки забезпечує корекцію некінематичних похибок, що виникають через термічні деформації, шум і похибки деформації елементів. Експериментальні дослідження, проведені на основі синтетичних даних, показали значне зниження середньоквадратичної похибки (MSE) координат вимірюваних точок та зменшення невизначеності вимірювань. Модель продемонструвала високу точність і стабільність, що підтверджує її ефективність для контролю геометричних параметрів енергетичного обладнання.*
Keywords: штучна нейронна мережа, зворотне поширення помилки, компенсація похибок, інформаційно-вимірювальна система, координатно-вимірювальна рука, геометричні параметри, моделювання.

Надійшла до редколегії: 25.12.2024