conservation, thermal safety and envirimental pollution control (ICBTE 2020) October 29 – 30, 2020, Brest, Belarus E3S Web Conf. Volume 212, 2020 https://doi.org/10.1051/e3sconf/202021202002.
18. Barsukov, V.G. Vliianie defektov mikrostruktury na svoistva kompozitnoi stroitel'noi armatury / V.G. Barsukov, S.A. Sazon, E.G. Gniadek, A.Iu. Nosko // Vesnik GrDU imia Ianki Kupaly. Ser. 6, Tekhnika. – 2021. – T. 11. – №2. – S. 102 – 111.
19. Stavrov, V.P. Mekhanika kompozitnykh materialov - Mn.: BGTU, 1996. – 165 s.
20. Barsukov, V.G. Raschetnaia otsenka termouprugikh napriazhenii v okrestnostiakh volok-nistykh i dispersnykh napolnitelei v kompozitakh na osnove ftoroplastov /V. G. Barsukov, Iu. M. Dal', R. A. Lubovskaia, V. V. Barsukov // Vesnik GrDU imia Ianki Kupaly. Ser. 6, Tekhnika.– 2015. –№2 – S. 6 – 13.
21. Vliianie khimicheskoi usadki sviazuiushchego v protsesse otverzhdeniia na obrazovanie ostatochnykh napriazhenii v tsilindricheskikh obolochkakh iz kompozita / Iu. V. Vasi-levich [i dr.] // Teoreticheskaia i prikladnaia mekhanika: mezhdunarodnyi nauchno-tekhnicheskii sbornik. – Vyp. 31. – 2016. – S. 67 – 72.
22. Stavrov, V. P. Mekhanika kompozitnykh materialov – Mn.: BGTU, 1996. – 165 s.
23. Ken'ko, V. M. Nemetallicheskie materialy i metody ikh obrabotki / V. M. Ken'ko. – Minsk: Dizain PRO, 1998. – 240 s.
24. Minenkov, B. V. Prochnost' detalei iz plastmass / B. V. Minenkov, I. S. Stasenko – M.: Mashinostrenie, 1977. – 264 s.

UDC 624.012

FULL CONNECTED NEURAL-NETWORK FOR SIMULATION OF EXTANTION IN SELF-STRESSED MONOLITIC SLABS ON GROUND

Zheltkovich A. E. [1], ***Molosh V. V.*** [2], ***Parchotz K.G.*** [3], ***Saveiko N.G.*** [4], ***Yuan Jinbin*** [5], ***Zhenhao Jiang*** [6], ***Zheng Haoyuan*** [7]

[1]*Ass. Prof, PhD, Ass. Prof, Brest State Technical University, Belarus, e-mail: gelpek@mail.ru*
[2]*Ass. Prof, PhD, Ass. Prof, Brest State Technical University, Belarus, e-mail: m.vic@rambler.ru*
[3] *Programmer engineer, Belarus, e-mail: konstantinparhoc@gmail.com*
[4] *Programmer engineer, Belarus, e-mail: nick-2009@live.ru*
[4] *Postgraduate, Belarusian National Technical University, Belarus, e-mail: jinbinyuan@gmail.com*
[6] *Postgraduate, School of Data Science, The Chinese University of Hong Kong in Shenzhen, China, e-mail: 222041010@link.cuhk.edu.cn*
[7] *Postgraduates, Dalian University of Technology-Belarusian State University Joint Institute, China-Belarus, e-mail: 879777236@qq.com*

Abstruct

In this article the strategy of interdisciplinary convergence of mechanics and artificial intelligence is illustrated. The article presents the results of calculating displacements in self-stressed monolithic slabs on ground obtained using a trained fully connected neural network. The empirical results of displacements in slabs on ground, displacements calculated according to the physicomechanical model, and obtained using

a neural network are represented. The inspiration brought us to study neural networks modeling biological neural networks are follow: neural networks can autonomously detect patterns hidden in phenomena and can identify parameters on complex behavioral tracks of different physical systems. The authors describe in detail the developed and trained fully connected neural network.

Key Words: Artificial Neural Networks, Deep Learning Algorithm, Neurons, Slabs on ground. Self-Stressed Concrete.

МОДЕЛИРОВАНИЕ ПЕРЕМЕЩЕНИЙ В САМОНАПРЯЖЁННЫХ МОНОЛИТНЫХ ПЛИТАХ НА ОСНОВАНИИ ПРИ ПОМОЩИ ПОЛНОСВЯЗНОЙ НЕЙРОННОЙ СЕТИ

Желткович А. Е., Молош В. В., Пархоц К. Г., Савейко Н. Г., Юань Цзиньбинь, Чжэньхао Цзян, Чжэн Хаоюань

Реферат

В статье проиллюстрирована стратегия междисциплинарной конвергенции механики и искусственного интеллекта. Представлены результаты расчёта перемещений в самонапряжённых монолитных плитах на основании полученные с использованием обученной полносвязной нейронной сети. В данной статье представлены эмпирические результаты перемещений в плитах, перемещения, рассчитанные по физико-механической модели и полученные с использованием нейронной сети. Интерес, проявленный к нейронным сетям, моделирующим биологические нейронные сети, вызван тем, что, нейросеть может автономно обнаруживать закономерности, скрытые в феноменах, нейронные сети могут идентифицировать параметры на сложных поведенческих треках физических систем. Авторы подробно описывают разработанную и обученную полносвяз-ную нейронную сеть.

Ключевые слова: искусственные нейронные сети, алгоритм глубокого обучения, нейроны, плиты на основании, самонапряженный бетон.

Existed methods of calculation of deformations in slabs on ground

The areas of application of self-stressed reinforced concrete are constantly expanding. It is especially effective to use such concretes in structures and structures that have high requirements for cracking resistance, water penetration resistance and durability. Stressed concrete is used quite successfully in aerodrome and road construction [1]. V. Mikhailov [2] shows that to increase the length of the continuous section of the slab, it is necessary to use stressed concrete on cement with a self-stress grade of at least 3.0 N/mm^2.

Another of the most interesting areas of application of concretes with an adjustable level of self-expansion (for a shrinkage compensation) is the covering of floors of industrial buildings, in particular, floors of meat and dairy industry enterprises. There are a number of industries where it is particularly preferred to use self-stress concrete as the surface layer.

For example, during operation, concrete and reinforced concrete structures of such buildings are exposed to industrial aggressive media in the form of fatty acids, aqueous solutions of various salts, as well as temperature and humidity factors that cause intense corrosion of materials. Increased durability of such structures is achieved due to increased density of concretes, their strength, freeze-thaw resistance, impermeability and corrosion resistance. The listed effects are provided both due to the dense structure and the peculiarities of the chemical and mineralogical composition of the hardened cement paste of the expensive cement.

Together, as noted in [3, 4], it is necessary to clarify the nature of the distribution of bounded forced deformations and stresses along the length of the slab when the slab interacts with the base. At the same time, in [3, 4] it should be noted that none of the above studies, when determining the self-stress stain (SSS) of slabs, paid due attention to the consideration of the factor of interaction of the slab with the base (taking into account the restricting effect of the base when expanding concrete).

For example, in the work of prof. Mikhailov [2] value of SSS in characteristic sections along the length of the slab is proposed to be determined depending on the calculated value of coupling (bounded) deformation $\varepsilon_{CE}(x,t)$:

$$\sigma_{CE}(x,t) = 0.085 \cdot \left(f_{CE,d}\right)^{1.25} \cdot \left(\frac{1}{\varepsilon_{CE}(x,t)}\right)^{0.25} \tag{1}$$

where: $f_{CE,d}$ – grade of self-stress cement by energy activity, (kg/cm^2).

In models, where, one way or another, the interaction of the slab with the base was taken into account [5, 6], were revealed some shortcomings. The main of which was the fact of applying an analogy of temperature impact.

In the work [3], a physical-mechanical model of the SSS of the slab on ground was presented and the results of numerous laboratory and field tests of the slabs were given. Existing methods for calculating from expansion forced deformations, including those developed by one of the authors of the article within the framework of the Ph.D. dissertation, for certain time intervals, do not always assess the deformed state of the slabs with high accuracy. In [3], the displacements in slabs were determined based on the equilibrium equation recorded for the infinitesimal part of slab. A general differential equation describing the displacements of the slab has a view:

$$\frac{d^2u}{dx^2} - \frac{\tau(u)}{E_C \cdot h} = 0 \tag{2}$$

Solution of equation (2) with respect to displacements u, under the following initial conditions: $x = L/2$; $u = 0$; $x = 0$, is presented as:

$$u = -\frac{1}{\beta} \cdot \left(\frac{\sigma_{C0}}{E_{C,t}} + \varepsilon_0(t,t_0)\right) \cdot \frac{\sinh \beta\left(\frac{L}{2} - x\right)}{\cosh \beta\left(\frac{L}{2}\right)} \tag{3}$$

where: u – absolute displacement in the slab; $\varepsilon_0(t,t_0)$ – deformation of free expansion of concrete; $\varepsilon_{C,t}$ – elastic deformation of concrete; $E_{C,t}$ – modulus of elasticity of concrete, σ_{C0} – internal axial stress; $\sigma_{C0}/E_{C,t}$ – elastic component of relative deformations of concrete; β – coefficient depending on the characteristics of the contact of the slab with the base, the modulus of elasticity, the geometry of the plate.

At the same time, as practice has shown, the physical mechanical model, although it approached significantly to empirical data, but starting from certain time intervals (after 72 hours of hardening), demonstrates slightly overestimated results. In order to clarify the characteristics of SSS, we developed and trained (on the available experimental data) an artificial neural network (ANN) of a fully connected architecture with four incoming neurons, five hidden layers and eight neurons of the output layer. Each of the eight output layer neurons learned to "see" the displacements of slab at a specific coordinate. The illustration of fully connected layers applied in our research is shown in Fig. 1. On a Fig. 2 is represented program code has been written on Python. As an activation function of neurons, the «Relu» function inserted into the Python software language profile was used.

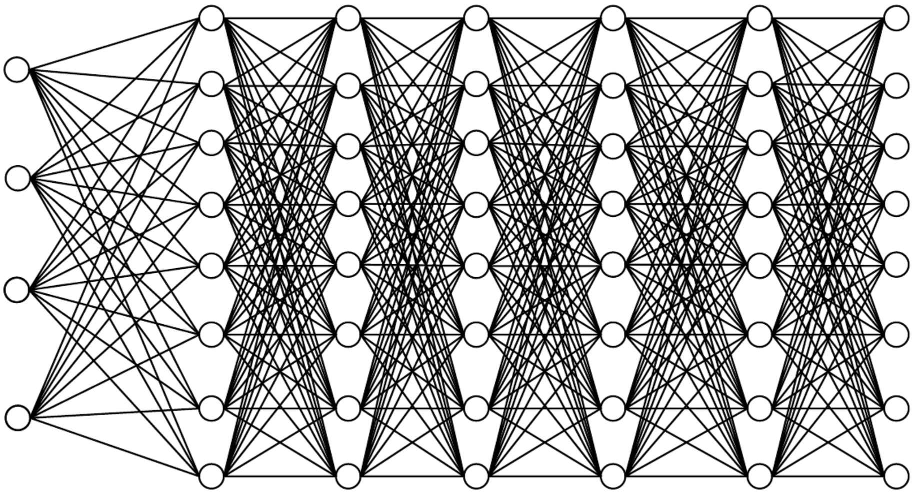
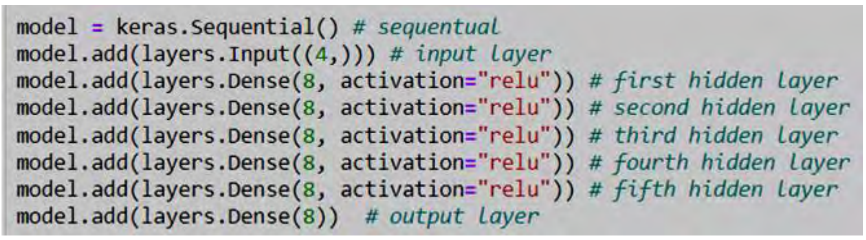


Fig. 1. The illustration of ANN with fully connected layers



```
model = keras.Sequential() # sequentual
model.add(layers.Input((4,))) # input layer
model.add(layers.Dense(8, activation="relu")) # first hidden layer
model.add(layers.Dense(8, activation="relu")) # second hidden layer
model.add(layers.Dense(8, activation="relu")) # third hidden layer
model.add(layers.Dense(8, activation="relu")) # fourth hidden layer
model.add(layers.Dense(8, activation="relu")) # fifth hidden layer
model.add(layers.Dense(8))  # output layer
```

Fig. 2. Realization on Python of fully-connected ANN with five hidden layers having eight neurons in each layer

Research Status of ANN for Parameter Recognition

Things in reality are extremely complex, and it is difficult for people to go deep into things to see the essence, and it is difficult to know which features are important. The inspiration brought to us by studying biological neural networks is that machines can autonomously discover the laws hidden in knowledge, rather than simply instilling knowledge into computers, and this ANN can identify parameters at complex behavioral interfaces.

Approach which employs computational models that mimic the architectural, structural, or functional aspects of biological neural networks, such as the human brain are known from the middle of the XX century.

Presently some scholars are proposing ANN to be widely used with deep learning model as an important part of the efficient target detection algorithm. ANN plays role of a feature extractor in the target detection algorithm. This scheme allows to computers autonomously capture experience from massive amounts of knowledge.

Mechanism of artificial neural network

Neuron is the basic element of ANN which usually includes two parameters. In ANN, there are several layers and each layer contains several neurons [7]. Each neuron can be considered as an operator which can change input data. The equation from input to output in a fully connected layer is expressed as follows:

$$Y = X*W+B \tag{4}$$

Where: *X* and *Y* are input and an output vector respectively, *W* is a weight matrix and *B* is a bias.

Although the computation is simple in neurons, many superimposed neurons have infinite possibility.

Full connection of neurons

In a brain, neutrons link with each other which are convenient for information transmission. In a neural network, full connection layer also means many links. In the fully connected layer, each neuron relates to all neurons in former layer and next layer [8]. It means that each neuron in fully connected layer can capture the global information of data. ANN can extract the underlying reflection from input to output with just addition and multiplication operation. So, a well-trained network can respond quickly without mechanical computation. Because the number of parameters of ANN is large, an efficient large-scale matrix solve method is needed. Gradient decent makes it possible to train a network easily. Besides, back propagation method is used to update the parameters [9]. To evaluate the parameters, loss function is introduced. Loss function is proposed to evaluate the gap between the true label (target) and the prediction given by neural network. Thus, the process of training neural network becomes an optimization problem which aim is to minimize the loss. After training network, a suitable evaluation metric is also important that can provide an objective comment.

Loss function

Loss function is designed to indicate the optimization direction of ANN.

In a regression task, the target is continuous which means that the value of label can sampled from a given interval. Traditionally, relative error is a suitable metric to evaluate the gap between true value and output of model. However, the training stage of ANN employs a concept of “batch”. Dataset is split into several batches to input the model and the optimization direction relies on the global information of one batch of data. Mean square error (MSE) is proposed to evaluate the batch error in regression task [10]. MSE can be formulated as following equation:

$$L = \frac{1}{n} * \sum \left| x_i - y_i \right|^2 \tag{5}$$

Where: *n* is the number of samples, *x* and *y* are prediction and target (actual data) respectively.

Luckily, relative error plays an important role in validation and test stage of ANN. Here will be used relative error to evaluate the trained model since it can intuitively reflect accuracy of the prediction result for single given sample.

Parameter optimization

Goal of parameter optimization is to find a minimum on hyper-surface of values of weights *W* which results in minimum values of loss function *L*. On each iteration algorithm updates weights W of parameters in neurons. To find the correct updating difference we need to calculate anti-gradient. Gradient points to the direction of the fastest function increase then to find minimum we need to take negative gradient – anigradient.

After calculating the loss the gradient information is transmitted from output layer to previous ones, layer by layer back to input layer, updating all layers, which is called back propagation BP [11]. The equation of iteration of parameter update is formulated as follows:

$$w_{i+1}=w_i-\alpha*\frac{\partial L}{\partial w_i} \tag{6}$$

Where: w is weight parameter, α is learning rate and L is loss.

With the help of gradient information, model can search relatively optimal parameters towards correct direction.

Regularization

For ANN, the training process can be considered as an optimization problem whose objective is to minimize the loss. However, training ANN does not pursue zero loss or global optimal solution (minimum). Because the search space is established on the training set (the sub-dataset used to train ANN) instead of validation (test) set. Because of the distribution of dataset, the best parameters for ANN on training set are always not the best on test set. Sometimes when ANN achieves an extremely outstanding performance on training set, it may have poor results on the test set. Model would pay more attention to some noisy samples that makes its structure complex. This process is called over-fitting or over-teaching [12] as shown on Fig. 3.

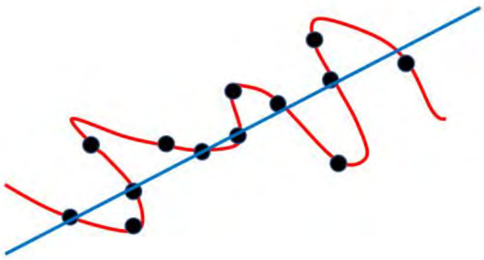


Fig. 3. Two fitting curves

So, it is necessary to avoid over-fitting. One of the most popular methods is regularization. The core idea of regularization is to restrict the complexity of model through controlling parameter to avoid fitting some special samples. Here we employ early stop strategy to prevent tuning to the training set too much.

Normalization

To help model in fitting the data we can implement different methods of data preprocessing. Standard score is one of them. Standard score sets mean of the set to zero and scale values based on standard deviation. Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual features do not look like standard normally distributed data. The normalization for x can be expressed as following equation.

$$x_{norm}=(x-\mu)/\sigma \tag{7}$$

Where: μ and σ are mean and standard deviation respectively.

Dataset generation

Fig. 4. – Determination of forced displacements in slabs on ground

On the basis of experimental data obtained during laboratory experiments, we generated a data matrix for 437 different time intervals. To form a matrix for the input and output layers of ANN, we have had the opportunity to use the following data obtained during physical experiments: average concrete compression strength – $f^{m}_{c,cube}$; Peak stress at the contact surface between the slab and the base – $\tau_{1,R(t)}$; Displacement corresponding to contact peak stress – $u_{1,R(t)}$ [3, 4]. Deformation of free expansion of concrete – ε_0; Self-stressing of concrete (equivalent to reinforcement bond in 1%) – σ_{CE}. A special .csv file was created for the data, as shown in Figure 4. In the second part of the matrix of the inputting ANN layer, data of absolute displacements of slabs obtained for nine points (coordinates) were placed. The displacements were determined by indicators set in 0.25 m increments along each of slabs [3]. In the program code, the number of input neurons was also limited to four, despite the fact that eight input parameters were originally provided in Figure 5.

```
data = pd.read_csv('Stress in Slabs on Ground Database//Deform in Slabs (actual) Database 29.08.2022 training.csv')
```

```
data.head()
```

	T	L	h	Rc	tmax	dmax	stressCE	freeexp	u0	u1	u2	u3	u4	u5	u6	u7	u8
0	19.0	2	0.1	4.00	0.294589	0.000013	0.210732	0.000163	0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.048000
1	19.5	2	0.1	4.08	0.294223	0.000013	0.216116	0.000170	0	0.0	0.000289	0.001053	0.001447	0.002316	0.002921	0.004421	0.055263
2	20.0	2	0.1	4.17	0.298694	0.000013	0.221493	0.000177	0	0.0	0.000579	0.002105	0.002895	0.004632	0.005842	0.008842	0.062526

Fig. 5. Data matrix [437 x 17] for input and output layers of ANN

Training artificial neural network

After preparation of the dataset, the next step is training of the artificial neural network. The whole step of training is implemented in Python (software) with Tensorflow library. Number of Epochs are set to 700, and 8% of the dataset has been used for validation of the results of the model, as shown in fig. 6.

```
history = model.fit(X_scaled, y_scaled, validation_split=.08, epochs=700)
Epoch 698/700
13/13 [==============================] - 0s 10ms/step - loss: 0.0097 - val_loss: 0.0040
Epoch 699/700
13/13 [==============================] - 0s 14ms/step - loss: 0.0096 - val_loss: 0.0089
Epoch 700/700
13/13 [==============================] - 0s 11ms/step - loss: 0.0100 - val_loss: 0.0061
```

Figure 6. Final steps of training ANN

When it comes to early stop strategy, training would finish early when MSE loss is lower than 0.01. Because network we used is light-weight, it needs little time to train. A loss curve is shown in Fig. 7. The lateral axis is iteration times and the vertical axis is the value of MSE loss.

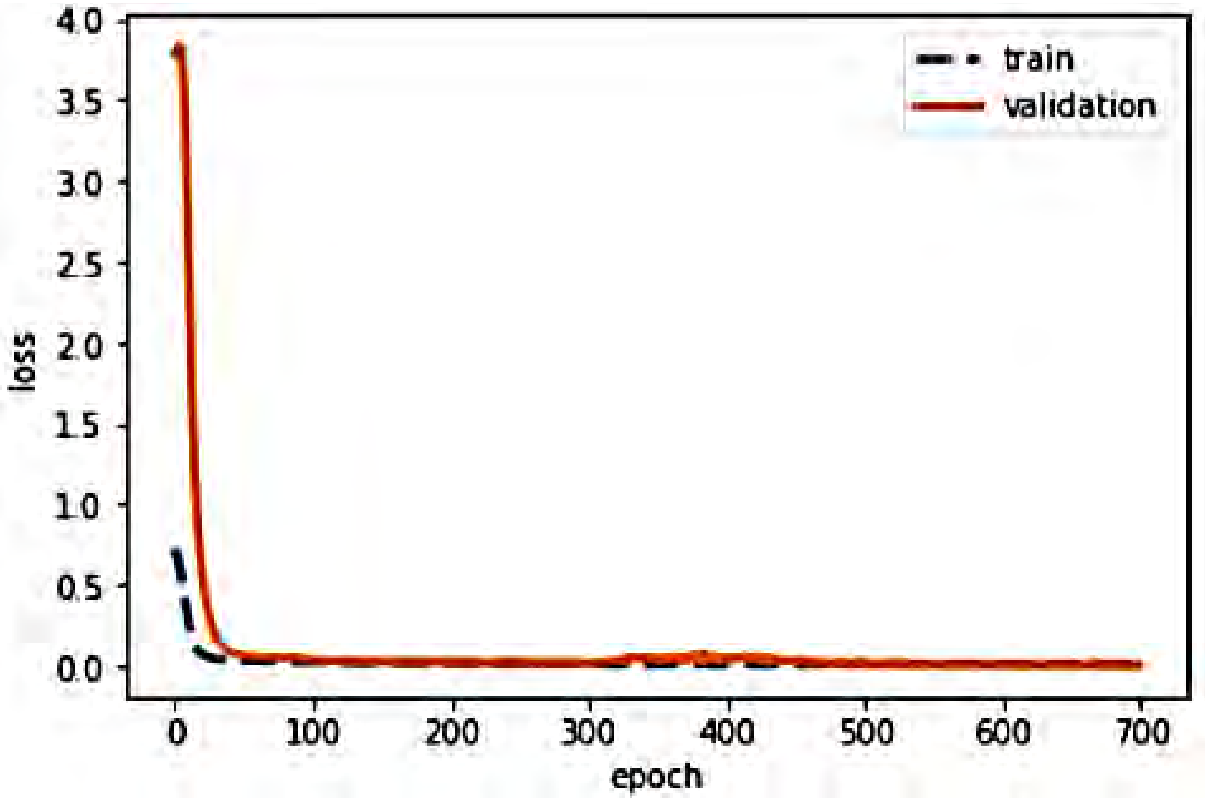


Fig. 7. Convergence and Loss Function

As shown in Figure 6, as the number of epochs increases, the MSE decreases continuously and finally converges.

During the training of the neural network, it turned out that the unchanged parameters (length and height of the slab), as well as the time parameter, and the free deformation of the expansion of concrete are often excessive and impair (worsen) the prediction accuracy (additional noise is created), so at this stage they were excluded.

Results and Evaluation

In the course of work on the program, a separate .csv file was formed with a test sample consisting of four samples taken for four time intervals. The test sample (matrix of input and output parameters) was determined for those time slices that were directly recorded in the independent laboratory experiment. The results are presented in Figure 8.

```
data_test = pd.read_csv('Stress in Slabs on Ground Database//Deform in Slabs (actual) Database 29.08.2022 test.csv')
```

	Rc	tmax	dmax	stressCE
0	4.16	0.297798	0.000013	0.367876
1	5.83	0.377431	0.000014	0.720705
2	7.00	0.425232	0.000015	1.073580
3	7.33	0.437937	0.000015	1.194000

Figure 8. Formed .csv file from a test sample

The diagrams below (fig. 9) show the results of ANN calculations of deformations in specific coordinates of slabs for time points corresponding to: 24. 47. 72 and 81 hours (displacement results are given in mm.) respectively.

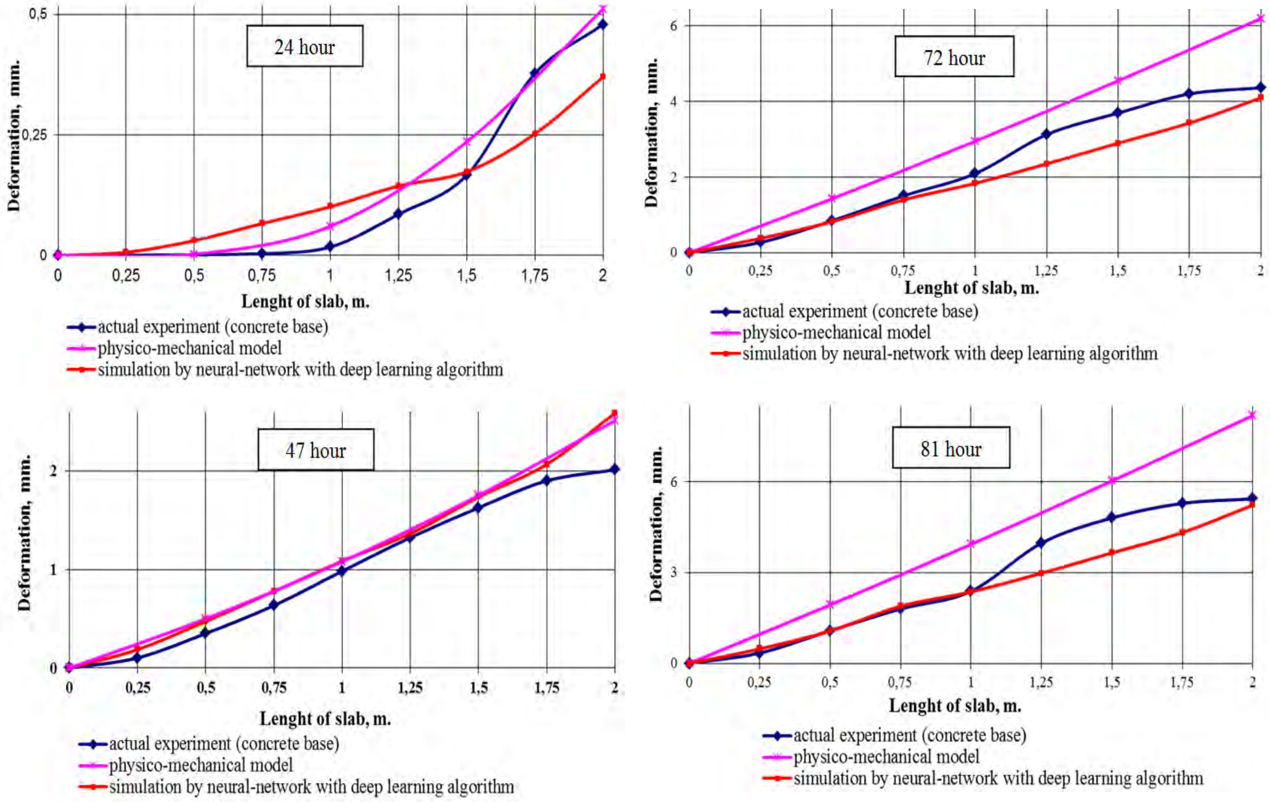


Fig. 9. Diagrams of displacements

To assess the reliability of the calculated models, the analysis of the ratios of the calculated with respect to (w.r.t) experimental values of slabs deformations, the correlation coefficient r_{ik} of the calculated and experimental values of deformations, as well as determined according to the procedure described in Appendix D to Technical Code of Common Practice EN 1990 [13], the correction coefficient b for the average value of the ratios u_{calc} / u_{exp} and the coefficient of variation of u_{δ} for the error vector δ (Fig. 10-11, Table 1) were used.

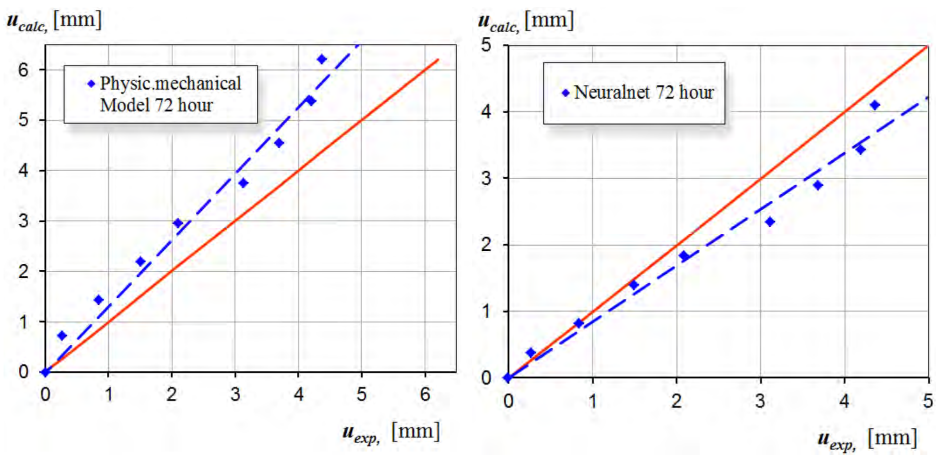


Fig. 10. Diagrams of variation of u_{δ} for the error vectors δ to 72 hours' time point

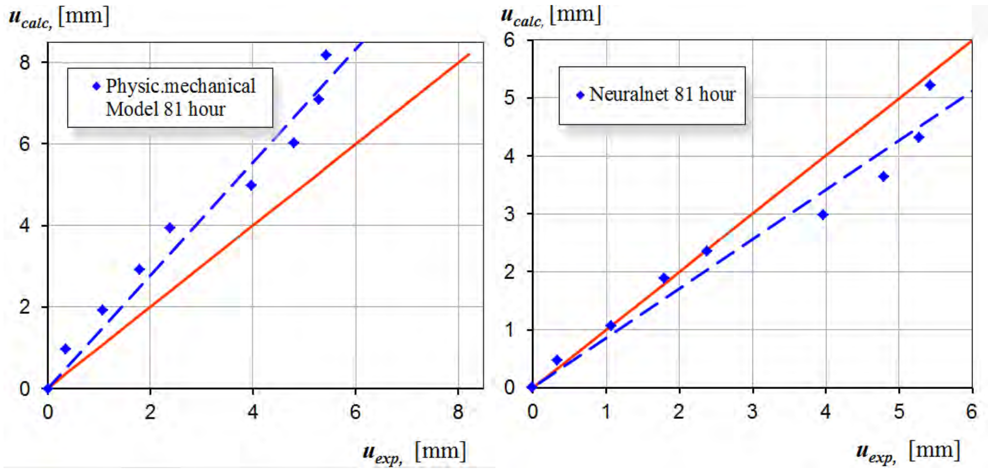


Fig. 11. Diagrams of variation of u_δ for the error vectors δ to 81 hours' time point

Table 1 - Statistical parameters of estimation of reliability of models

Results	*Min*	*Mean*	*Max*	C_u, %	*b*	u_δ, %	r_{ik}
24 hour							
Physico-mech. Model	0.989	2.362	8	1.02	0.909	0.126	0.988
Neuronetwork	0.667	5.467	16.75	1.089	1.182	0.228	0.957
47 hour							
Physico-mech. Model	1.069	1.336	2.404	0.335	0.865	0.177	0.992
Neuronetwork	1.03	1.244	1.808	0.204	0.868	0.119	0.988
72 hour							
Physico-mech. Model	1.2	1.535	2.6	0.298	0.754	0.043	0.991
Neuronetwork	0.752	0.931	1.382	0.213	1.174	0.032	0.989
81 hour							
Physico-mech. Model	1.253	1.656	2.823	0.308	0.713	0.174	0.986
Neuro-network	0.751	0.965	1.395	0.216	1.156	0.133	0.981
Min, mean, max - minimum, average and maximum values of u_{calc} / u_{exp} ratios, respectively; C_u - coefficient of variation of u_{calc} / u_{exp} ratios; *b* - correction factor for average value of relations u_{calc} / u_{exp}; u_δ – coefficient of variation for the error vector δ of u_{calc} / u_{exp} ratios; r_{ik} – correlation coefficient of experimental u_{exp} and calculated u_{calc} deformation values.							

Conclusions

1. In paper has been verified the feasibility of using trained ANN to identify physical parameters. It is not difficult to notice a significant improvement in the predictive power of the neural network w.r.t the physicomechanical model over time. C_u - coefficient of variation or u_{calc} / u_{exp} ratio for the moment of time - 81 hours, obtained by the neural network decreased by 30% w.r.t. the physicomechanical model. Coefficient of variation of the error vector decreased too (on 24%).

2. To improvement the proper design of slabs on ground, AI aided method is universal and promising. In addition, this makes it possible to promote prognostic methods of description of stress-strain state condition in such slabs. ANN with deep learning algorithm proved that it can be implemented in other engineering problems. If the data

richness of the training set continues to increase, the prediction accuracy of this neural network will be further improved.

3. In next steps, the predicting the SSS of monolithic slabs on ground, with using of genetic algorithms of various complexity and neural networks of different architectures will be worked out.

References

1. Aleksandrovsky S.V., Bakum P.F. Pre-stressed and self-stressed concrete in the USA [*Predvaritelno napryjennii I samonapryjennii beton v SShA*] / M.: Stroyizdat, 1974. - 320 p.

2. Mikhailov V. V., Litver S. L. Expanding and stressing cements and self-stressed constructions [*Rasshiriaiushchie i napriagaiushchie cementy i samonapriazhennye konstruktsii*]. Moscow, Stroyizdat, 1974.

3. Zeltkovich A. E. Stress-strain state of monolithic slabs interacting with the base at the stage of expansion and shrinkage of concrete [Napryjenno-deformirovannoe sostoynie monolitnih plit, vzaimodeistvuiushchih s osnovaniem na stadii rasshireniy I usadki betona]: abstract of the dissertation... cand. teh. sciences: 05.23.01 / BSTU. – Brest., 28 p.

4. Zeltkovich A. E., Tur V. V. Calculation of displacement and stresses from shrinkage of monolithic concrete slabs that interact with the ground [*Raschet vynuzhdennykh peremeshchenii i napriazhenii ot usadki v monolitnykh betonnykh plitakh, vzaimodeistvuiushchikh s osnovaniem*]. Building science and technology. 2011, №. 2, pp. 120–125

5. Pettersson D. Stresses in concrete structures from ground restraint, Licentiate thesis. – Stockholm, – 1998. Part I. – 58 p.

6. Pettersson D. Stresses in concrete structures from ground restraint, Licentiate thesis. – Stockholm, – 1998. Part II. – 63 p.

7. Asteris P.G, Mokos V.G. Concrete compressive strength using artificial neural networks [J]. Neural Computing and Applications, 2020, 32(15): 11807-11826.

8. Sun W, Ma S, He X, et al. SimpleMeshNet: end to end recovery of 3d body mesh with one fully connected layer [J]. Journal of Real-Time Image Processing, 2022:1-11.

9. Shin H.C, Roth H.R, Gao M, et al. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning [J]. IEEE transactions on medical imaging, 2016, 35(5): 1285-1298.

10. Laxmi S.B, Vijaya M.S. Weighted Mean Square Error Technique to Train Deep Belief Networks for Imbalanced Data [J]. International Journal of Simulation-Systems, Science & Technology, 2018, 19(6).

11. Chen H, Lu F, He B. Topographic property of backpropagation artificial neural network: From human functional connectivity network to artificial neural network [J]. Neurocomputing, 2020, 418: 200-210.

12. Poernomo A, Kang D K. Biased dropout and crossmap dropout: learning towards effective dropout regularization in convolutional neural network [J]. Neural networks, 2018, 104: 60-67 pp.

13. Basis of Design of Building Structures [Asnovy Proektavannya Budaunichy Kanstruttsy]: ТКП EN 1990-2011. - Anavar. 15.11.2012. - Minsk: CEN/TC 250 "Structural Eurocodes": Ministry of Architecture and Construction of the Republic of Belarus, 2011. - 70 s.