JOURNAL OF CIVIL ENGINEERING AND MANAGEMENT
ISSN 1392-3730 print/ISSN 1822-3605 online
2013 Volume 19(5): 696–704
doi:10.3846/13923730.2013.799091

OPTIMIZATION OF MULTISPAN RIBBED PLYWOOD PLATE MACROSTRUCTURE FOR MULTIPLE LOAD CASES

Jānis ŠLISERIS, Kārlis ROCĒNS

Department of Structural Engineering, Riga Technical University, Kaļķu street 1, LV-1658 Riga, Latvia

Received 20 Sep. 2011; accepted 21 Dec. 2011

Abstract. This paper discusses an optimized structural plate of plywood composite that consists of top and bottom plywood flanges and a core of plywood ribs. The objective function is structure's weight. Typical constrains – maximal stress criteria and maximal deformation criteria – are used. The optimization is done by Genetic Algorithm (GA), and optimization results are used to train Feed-Forward Artificial Neural Network. The numerical simulation of plywood structure is done by using classical linear Kirchoff–Love theory of multilayer plate and Finite Element Method. As a result, an effective optimization methodology for plywood composite material is proposed. The most rational (according to strength-stiffness criteria) plywood composite macrostructure is obtained for some typical cases.

Keywords: plywood composite; structural optimization; Genetic Algorithm; Artificial Neural Network.

Reference to this paper should be made as follows: Šliseris, J.; Rocēns, K. 2013. Optimization of multispan ribbed plywood plate macrostructure for multiple load cases, *Journal of Civil Engineering and Management* 19(5): 696–704. http://dx.doi.org/10.3846/13923730.2013.799091.

Introduction

Optimization of plywood macrostructure that provides minimal material consumption, weight, and good heat–moisture performance is one of the most challenging tasks in today's engineering science. Plywood structures provide a rational usage of wood (Sliseris, Rocens 2010); one of the reasons, for instance, is less sawdust in manufacturing. The maximal thickness of typical hot-pressed plywood sheet is limited and, in many cases, there is insufficient bending stiffness. The object of this research is plywood composite that consists of top and bottom plywood flanges and a core that is formed from plywood ribs.

As a result of the structure's complexity (many variable parameters or complicated relationships, many local minimums and/or maximums) between design vector and objective function, its macrostructure optimization is a very challenging problem. Therefore, classical optimization methods such as response surface method (Goremikins, Serdjuks 2010) may fail or find the local optimum (Rao 2009). The optimization of plywood macro-structure is achieved by the following optimization methods: Artificial Neural Network (ANN), Genetic Algorithm, and Simulated Annealing Method (Goldberg 1989; Sliseris, Rocens 2011). All these methods were inspired by processes in nature.

1. Plywood composite macrostructure

The optimization of macrostructure is done for plywood composite, and the cross section and structural analysis scheme of which are shown in Figure 1. This kind of scheme is widely used in the structure of heavy truck floor. The two point loads arise from the front wheels of forklift machine. The plywood composite consists of top and bottom plywood flanges with thickness t_1 and t_2, respectively. The core of the composite consists of plywood ribs with thickness t_3 (Fig. 1). The connection between flanges and ribs is obtained by a glued joint.

This kind of structure is characterized by good heat–moisture performance. The moisture caused swelling in transversal direction is significantly decreased by vertical ribs with most of wood fiber orientation in transversal direction. The voids in the structure provide better heat insulation properties especially if they are filled with foam-like insulation materials.

Corresponding author: Jānis Šliseris
E-mail: janis.sliseris@gmail.com

Taylor & Francis
Taylor & Francis Group

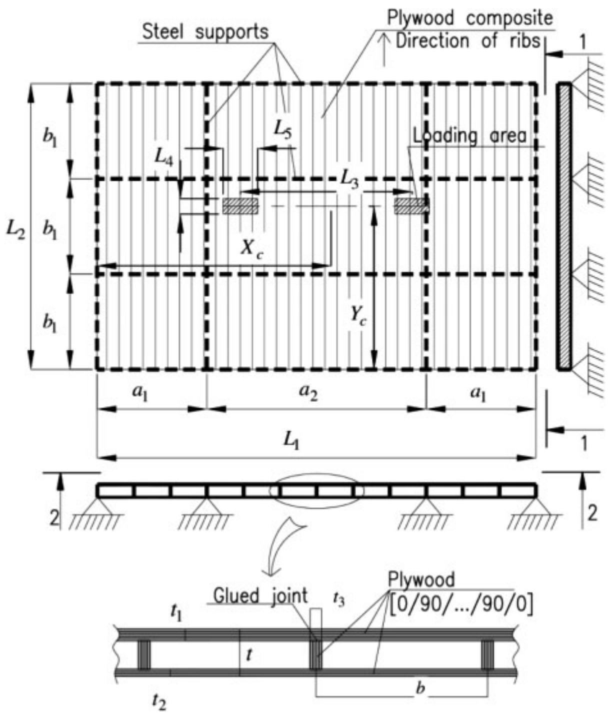

Fig. 1. The cross section and structural analysis scheme of plywood composite

The design vector consists of four parameters that characterize the cross section:

$$x = \{t_1, t_2, t_3, n\}, \tag{1}$$

where n is the number of ribs that is calculated by equation:

$$n = \frac{L_1}{b} + 1. \tag{2}$$

The supports are shown in Figure 1 by dashed lines. The load acting on the structure consists of two loaded areas with uniformly distributed load. The dimensions of loading areas are small compared to structure dimensions; therefore, they could be replaced by two point loads. Distance between the loading areas is $L_{3,}$ and the center coordinates of both loading areas are X_c and Y_c.

2. Optimization methodology

The plywood's composite macrostructure is optimized by the following algorithm:

- The main structure's properties – length L_1, width L_2, distance between steel supports – b_1, a_1, a_2, maximal thickness t, loading area properties: L_4, L_5, and load Q on each area are collected.
- The decision of which direction it is necessary to orient plywood ribs according to location of supports is made.
- The location (X_c, Y_c) of loading area that is the most dangerous (produce maximal stress and deflection) is obtained.
- The geometrical parameters $x = \{t_1, t_2, t_3, n\}$ of plywood's composite macrostructure by using structures total weight as an objective function are optimized.
- The previous steps are repeated for various main geometrical properties.
- The optimal Artificial Neural Network (ANN) architecture is chosen. The ANN training by using results from previous procedures is done.
- ANN testing.

It was decided to connect the Genetic Algorithm (GA) and ANN because they provide a relatively fast and precise optimization with good accuracy and rendering of optimal results for cases that were not optimized with GA.

The proposed methodology that is based on GA and ANN is universal and could be easily used to optimize a wide range of structures to obtain better physical–mechanical properties such as strength-stiffness, heat insulation, moisture swelling, sound insulation, and others. In case of other objective functions, constrains and design space should be changed.

3. Optimization methods

3.1. Genetic Algorithm (GA)

A general constrained minimization problem can be stated as follows:

$$\begin{cases} Minimize(f(x)) \\ g_i(x) \leq 0, i = 1..m \\ h_j(x) = 0, j = 1..p \end{cases}, \tag{3}$$

where: x – vector of design variables; $f(x)$ – objective function; $g_i(x)$, $h_j(x)$ – constrains that are in form of inequalities and equalities.

GA can be used for unconstrained problems. Therefore, Eqn. (3) is converted into an equal unconstrained minimization problem by using the concept of penalty functions as follows:

$$Minimize(\Phi(x)),$$

$$\Phi(x) = f(x) + \sum_{i=1}^{m} r_i (G_i(x))^2 + \sum_{j=1}^{p} R_j \Big(h_j(x)\Big)^2, \tag{4}$$

where r_i, R_j are penalty parameters, and its values are constant during optimization, $G_i(x)$ is defined as follows:

$$G_i(x) = \begin{cases} g_i(x), g_i(x) > 0 \\ 0, g_i(x) \leq 0 \end{cases}. \tag{5}$$

These penalty functions were chosen in particular because usually there are some constraints that are more significant and penalty parameters allow to

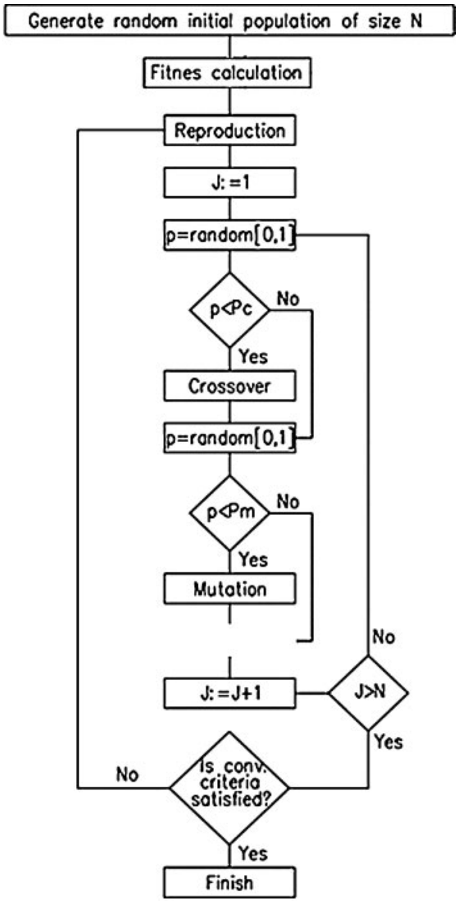

Fig. 2. The flowchart of GA

divide the constrains according to their significance. The square of functions $G_i(x)$, $h_j(x)$ allows avoiding negative values.

GA is based on the principles of natural genetics and natural selection (Darwin's theory of survival of the fittest). The basic operators of natural genetics are Reproduction, Crossover, and Mutation (Goldberg 1989; Sliseris, Rocens 2011). A simplified flowchart of GA is shown in Figure 2.

At the beginning of the algorithm, there is a generation of initial population that consists of generation of design vectors and calculation of its fitness functions. The genetic operators – Reproduction, Crossover, and Mutation are applied to initial population to generate new design vectors and eliminate the bad ones.

Reproduction is the first operator applied to the population to select "good" design vectors that gives a minimal value of objective function.

Crossover is the second operator applied to the population with initially defined probability Pc. The Crossover operator randomly selects two design vectors (called parents) from the population and by changing its binary codes obtains two new design vectors although there are also methods that use only decimal numbers (Rao 2009).

The Crossover operator generates random integer number in an interval from 1 to j, where j is the length of the binary code. By changing binary numbers of parents from i-th place, where i is generated as a random number, it obtains two new binary codes (called child). The Crossover operation is done only in case if the child gives a better value of fitness function.

Mutation operator is applied to the new binary codes with a specific small probability Pm. This operator changes each number in the binary code from value 1 to 0 or 0 to 1 with probability Pm, that is very small, usually $Pc < 0.01$.

3.2. Artificial Neural Network (ANN)

ANNs are inspired by biological nervous systems (Garson 1991). With many simplifications, the biological nervous system could be reduced to a set of cell bodies covered by membranes, dendrites, synapses, and axons. Around each cell body there are a set of dendrites that are connected to cell bodies by synapses and may be connected to other cell bodies by axons. Each dendrite sends an electrochemical signal to a cell body and if the sum of signals is strong enough, then the cell body potential exceeds a threshold and a pulse is generated and send by axon further to the other neuron. The total number of neurons in human brain is approximately 10^{11}, although in this work the mathematical model contains only 20 neurons.

ANN (mathematical model) have been trained to perform complex tasks in various fields (Goh 1995; Maier, Dandy 2001), including function fitting, pattern recognition, identification, classification, speech, vision, and control systems. In structural engineering, ANN is used widely in geotechnical analysis (Das, Basudhar 2006; Ferentinou, Sakellariou 2007; He, Li 2008).

ANN consists of an input parameter p, strength of scalar w, bias b, and an activation function f. The output scalar of neuron could be calculated by the following equation:

$$a = f(w \cdot p + b). \quad (6)$$

In Figure 3 is shown the ANN that consists of three layers. Input vector p contains R entities. The number of neurons in the first, second, and third layers are s^1, s^2, and s^3, respectively. Similarly the bias, activation function, and output: b^1, b^2, b^3, f^1, f^2, f^3 and a^1, a^2, a^3, respectively. The weight matrix in the first layer is $IW^{1,1}$(matrix dimensions $s^1 \times R$), in the second layer – $LW^{2,1}$(matrix dimensions $s^2 \times s^1$) and in the third layer – $LW^{3,2}$ (matrix dimensions $s^3 \times s^2$). In a more general form, $LW^{i,j}$ is the weight matrix that is used to calculate the output vector of j-layer and input vector of i-layer.

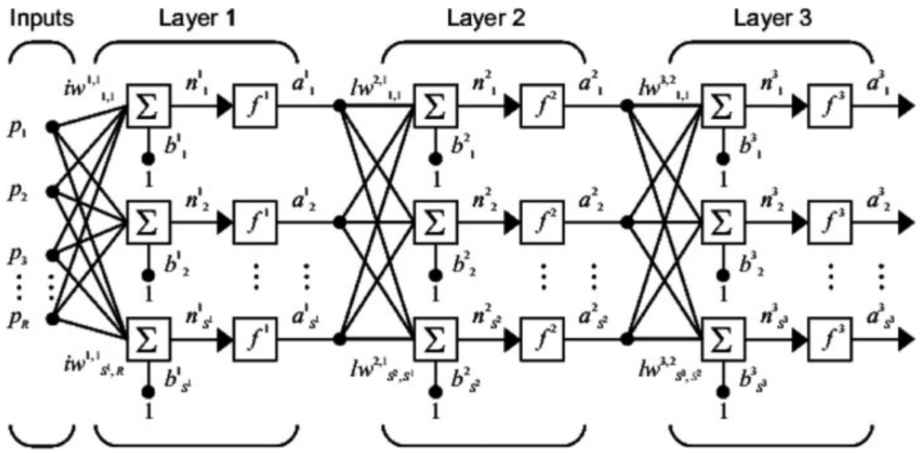

Fig. 3. Multilayer neural network architecture

The output vectors a^1, a^2, a^3 of ANN are shown in Figure 3. They could be calculated by the following equations:

$$a^1 = f^1\left(IW^{1,1} \cdot p + b^1\right); \tag{7}$$

$$a^2 = f^2\left(LW^{2,1} \cdot a^1 + b^2\right); \tag{8}$$

$$a^3 = f^3\left(LW^{3,2} \cdot a^2 + b^3\right). \tag{9}$$

After algebraic substitutions the Eqn. (9) changes to

$$a^3 = f^3\left(LW^{3,2} \cdot f^2\left(LW^{2,1} \cdot f^1\left(IW^{1,1} \cdot p + b^1\right) + b^2\right) + b^3\right). \tag{10}$$

Transfer function could be arbitrary but usually *purelin* (Fig. 4) or *tansig* (Fig. 5) function is used.

3.2.1. Training algorithms of ANN

The training process requires a set of examples – inputs p and target outputs t to provide the proper ANN behavior of ANN. Weights and biases of the ANN are iteratively adjusted to minimize the ANN performance function during the training process. The performance function is usually mean square error *mse* – the average squared error between the ANN outputs a and the target outputs t.

The simplest learning algorithm is back propagation. It updates the vector containing weight and bias matrix coefficients – X. It is done by using a gradient of the performance function. The vector X components are changed in the direction of maximal negative gradient of performance function so that it decreases most rapidly. It is described by the following equation:

$$X_{k+1} = X_k - \alpha_k \cdot g_k, \tag{11}$$

where: X_{k+1} – are the updated values of unknown vector; X_k – the old values of an unknown vector; α_k – the learning rate; g_k – the gradient of performance function.

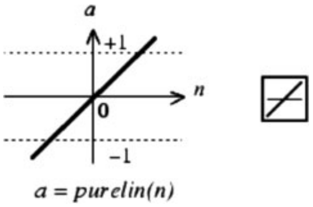

Fig. 4. Plot of *purelin* transfer function and its graphical sign

This algorithm is quite simple, but its convergence rate is slow. Next algorithms provide the convergence increase about 10 to 100 times.

Conjugate Gradient Algorithms (Hagan *et al.* 1996) are based on the fact that the function decreases most rapidly along the negative of gradient but this does not necessarily produce the fastest convergence. So the search is performed along the conjugate directions.

Quasi–Newton Algorithms are used very often (Hagan *et al.* 1996) that they often converge faster than Conjugate Gradient Algorithms. These algorithms are based on the following equation:

$$X_{k+1} = X_k - A_k^{-1} g_k, \tag{12}$$

where A_k^{-1} is the inverse of Hessian matrix (second derivatives of performance function with respect to vector X unknown coefficients).

The next algorithm is called Levenberg–Marquardt Algorithm (Hagan, Menhaj 1994) that usually works faster than all the earlier mentioned ones; therefore, it is used in this work. The Hessian matrix

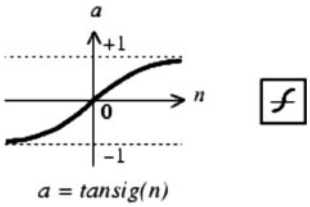

Fig. 5. Plot of *tansig* transfer function and its graphical sign

is approximated by the square of Jacobian matrix – J (it is true if the performance function is mean square error):

$$A = J^T \cdot J. \quad (13)$$

The vector X is updated according to the following equation:

$$X_{k+1} = X_k - \left[J^T J + \mu \cdot I\right]^{-1} \cdot J^T \cdot e, \quad (14)$$

where: e is the vector of mean square error *mse*; μ – scalar; I –identity matrix.

When $\mu \to 0$, then this method works like Quasi–Newton Method, but when it is large, then it works like Conjugate Gradient Method. The scalar μ is decreased after each successful step.

4. Obtained results

The optimization results are shown for a particular example with the following properties: $L_1 =$ 2.16 m, L_2 = 1.36 m, b_1 = 0.453 m, a_1 = 0.54 m, a_2 = 1.08 m, L_3 = 0.9 m, L_4 = 0.08m, L_5 = 0.18 m, Q = 2700 kg (on each load area, see Fig. 1). These dimensions and loads are commonly used in the structure of a heavy truck floor.

4.1. Analysis of the worst location of load

The coordinates of center of loading area – X_c and Y_c – were varied. The maximal stress in the direction of wood fiber and maximal deflection were obtained for each case. According to the results, the most dangerous (provides maximal stress and deflection of plate) location of load areas was obtained.

In Figures 6 and 7 are shown the relative-maximal deflection (ratio of U_z/U_z (max), where U_z – out of plane deflection) and in Figure 8 – relative-maximal stress in the direction of wood fiber depending on the coordinates of center of loading area. In Figure 9 is shown the most dangerous loading case (because of structures symmetry there are other three loading cases with the same dangerousness). The numerical simulation is done by commercial Finite Element Program Code ANSYS using PC with properties: Intel Core 2 Duo CPU 2.93 GHz, 2.5 GB of RAM.

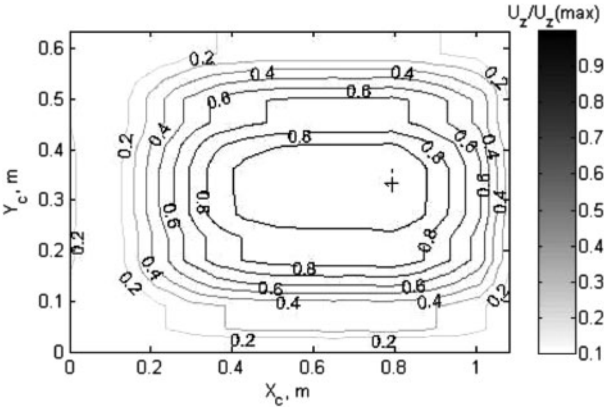

Fig. 7. One quarter part of the plot of maximal deflection, depending on center coordinates (X_c, Y_c) of the loading area

The worst-load case is obtained for specific bending stiffness properties. It is assumed that bending stiffness in Y direction is 15% greater than in X direction. This assumption was used because the actual bending stiffness depending on design vector is close to this assumption. There was done another simulation with a different ratio of bending stiffness and the location of worst-loading area was not changed significantly.

4.2. Optimization of plywood composite macrostructure

The optimization is done by using GA described in Section 4.1. GA is a stochastic algorithm, therefore, there was done three independent runs for each case. If the dispersion of results is less than 5%, then the best solution is taken. If the dispersion is more than 5%, then several extra runs are done.

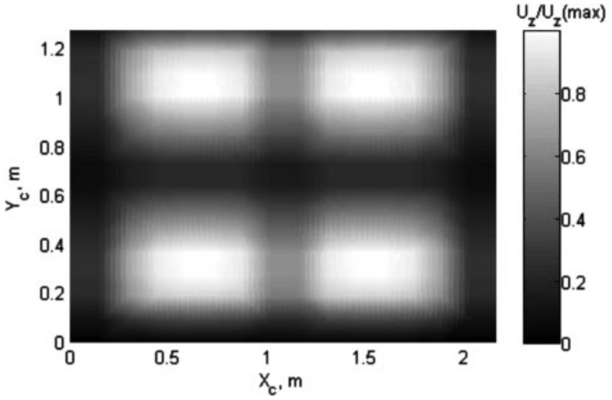

Fig. 6. Plot of relative-maximal deflection, depending on center coordinates (X_c, Y_c) of the loading area

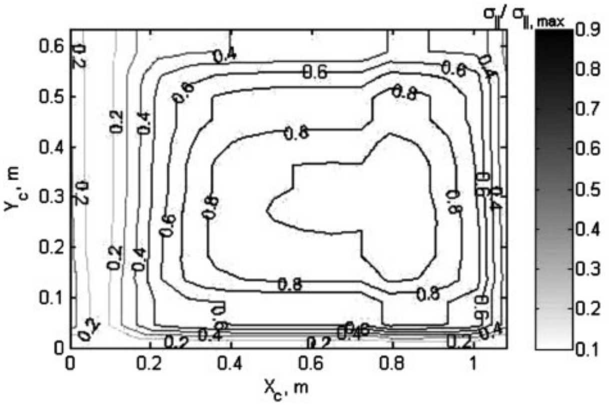

Fig. 8. The maximal stress in the direction of wood fiber, depending on the coordinates of the loading area center

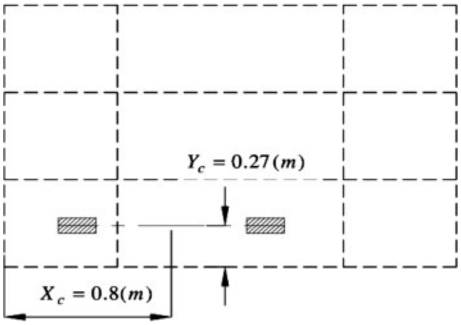

Fig. 9. The most dangerous location of the loading area center

The design vector (1) was optimized for various loads $Q = \{2; 2.5; 3; 3.5\}$ (tons per each loading area) and for various input geometrical parameters:

$L_1 = \{1.5; 2; 2.5; 3\}$ (m);
$L_2 = \{0.75; 1; 1.25; 1.5\}$ (m);
$t = \{30; 40; 50; 60\}$ (mm).

The total number of cases that were optimized is 256. Some of the results (in case if Q = 2.5 tons and t = 40 mm) are shown in Table 1. Constrains were calculated according to European Committee for Standardization (2004):

1) Stress in compressed flange;
2) Stress in tensioned flange;
3) Shear stress in plywood ribs;
4) Shear stress in glued joints;
5) Local bending of top plywood flange.

The plywood structure was modeled using linear plate Finite Element based on Kirchoff–Love theory of thin plates. The time of simulation was significantly decreased (compared to the full model, where plywood sheets are modeled with Shell Finite Elements) by dividing the calculation into two parts. In the first part, the plate was modeled by using homogenization method. It means that ribs were replaced with the same stiffness elastic layer. In the second part was simulated only the behavior of top flange. The ribs were assumed to be supports. The second part is necessary to model the local bending behavior of top plywood flange.

The plywood elements were simulated using the mechanical characteristics of Finnish birch plywood according to the *Handbook of Finnish Plywood*

Table 1. The geometrical parameters of optimized structure

L, m	*B*, m	t_1, m	t_2, m	*t*, m	*n*	*F*, kg
1.50	0.75	0.0130	0.0110	0.0200	15	115.3
1.50	1.00	0.0132	0.0115	0.0220	21	201.9
1.50	1.25	0.0139	0.0125	0.0250	28	350.4
1.50	1.50			Not satisfied constrains		
2.00	0.75	0.0136	0.0120	0.0250	14	120.5
2.00	1.00	0.0142	0.0127	0.0260	20	207.6
2.00	1.25	0.0146	0.0130	0.0300	27	353.8
2.00	1.50			Not satisfied constrains		
2.50	0.75	0.0143	0.0128	0.0280	14	126.6
2.50	1.00	0.0148	0.0130	0.0300	20	215.0
2.50	1.25	0.0150	0.0130	0.0300	26	361.6
2.50	1.50			Not satisfied constrains		
3.00	0.75	0.0150	0.0130	0.0300	14	133.1
3.00	1.00	0.0150	0.0130	0.0300	20	223.2
3.00	1.25	0.0150	0.0130	0.0300	30	401.0
3.00	1.50			Not satisfied constrains		

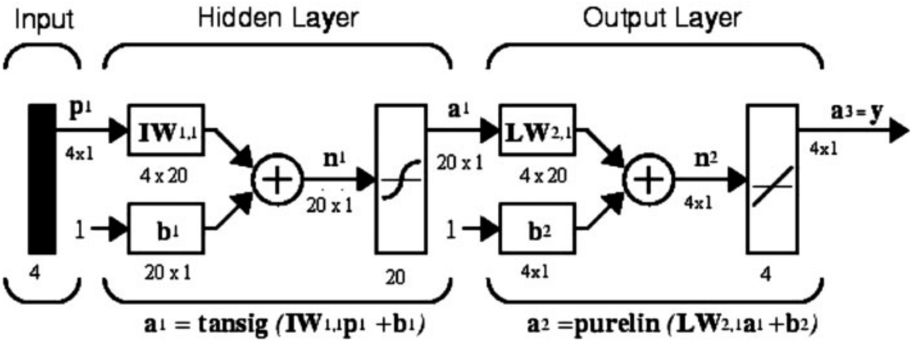

Fig. 10. Feed-Forward ANN with one hidden layer and one output layer that was trained

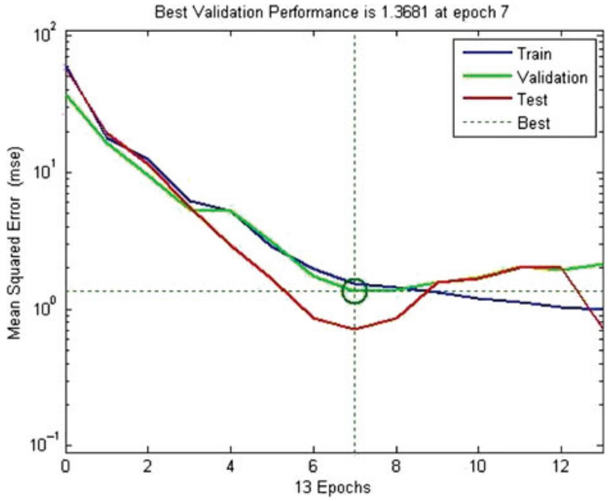

Fig. 11. Validation plot, mean square error, depending on training epoch

(Finnish Forest Industries Federation 2002). As we can see in Table 1, there are some cases when the constrains are not satisfied. For example, in the case when $L = 1.5$ and $B = 1.5$ m, the constraint of stress in compressed flange is not satisfied, therefore, the feasible solution was not achieved. Because of these cases, there arise problems in ANN training process. The ANN should calculate optimal parameters and also warn about the cases when it is not possible to satisfy the constrains. We solved this problem by modifying training data by changing all the components of design vector to zero in case if constrains are not satisfied. The user of ANN could identify those cases when neural network gives values of optimal parameters very close to zero.

4.3. Artificial Neural Network

The data that were obtained in the optimization is used to train the ANN (totally 256 cases). The input vector consists of four parameters $p = \{Q, L_1, L_2, t\}$. Output vector consists of four parameters $y = \{t_1, t_2, t_3, n\}$. The numerical experiments show that the minimal size of ANN is 20 neurons. It is necessary for accurate simulation of this size of data. The architecture of ANN that was used in the analysis is shown in Figure 10. The transfer function in the hidden layer is *tansig* function (Fig. 4), but in the output layer, it is *purelin* function (Fig. 5).

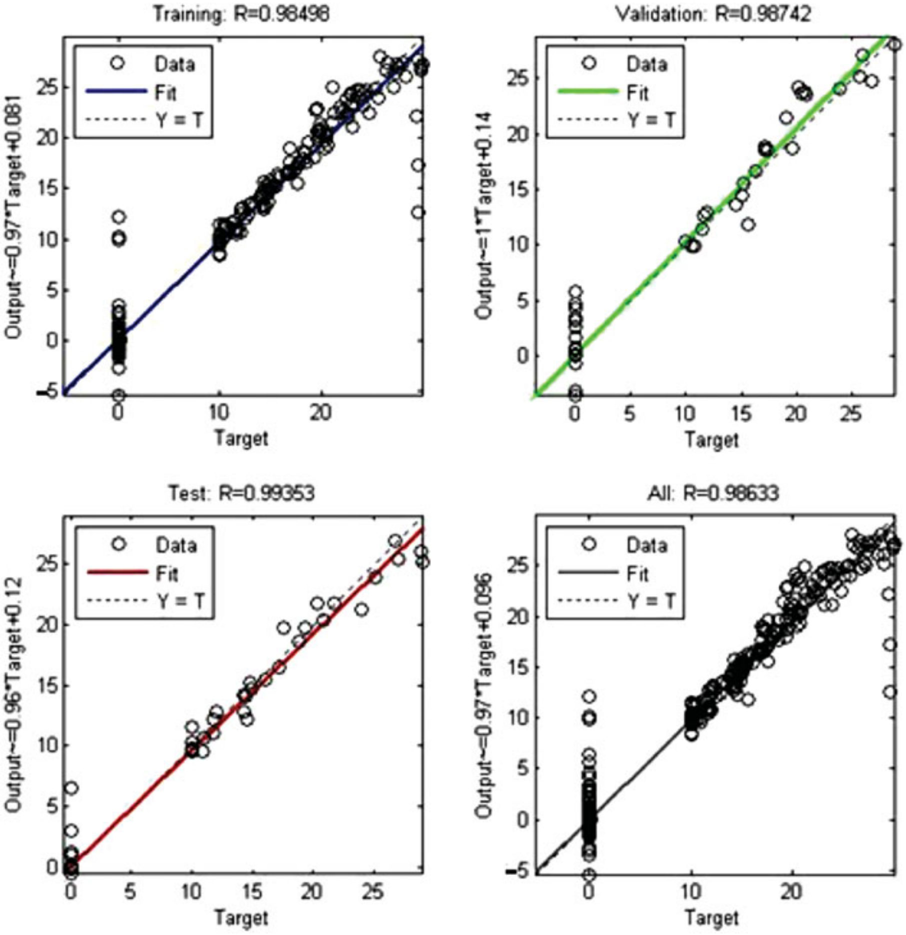

Fig. 12. The regression plots of output of ANN and target results

The ANN was trained by using Levenberg–Marquardt Algorithm that is based on Eqn. (14). This is the most appropriate algorithm for Feed-Forward Artificial Neural Network for many practical cases (Hagan *et al.* 1996).

Validation and testing of ANN were simultaneously done during the training process (in each training epoch). The validation shows that the optimal number of training epoch is 7 (see Fig. 11, green curve). The mean square error at the epoch number 7 is nonsignificant (see Fig. 11, red curve). If the ANN is trained more than 7 epoch, then it is overtrained and mean square error increase (see Fig. 11, green curve).

Regression analysis of the ANN was done to obtain accuracy of results. The regression plot of optimized parameter n (number of plywood ribs) is shown in Figure 12. Very similar plots are obtained for other output parameters. The analysis of the correlation coefficient R shows that it is very close to 1 ($R > 0.98$). It means that there is strong correlation between ANN output and the target results. There are problems with the cases when constrains are not satisfied and components of ANN output vector should be very close to zero. Figure 12 shows that sometimes ANN could not identify correctly the cases when constrains are not satisfied.

Neural Network and Genetic Algorithm Toolbox of Matlab 7.6.0 software were used in the optimization and training of ANN. Special procedures that were necessary in analysis are written in Matlab 7.6.0 programming environment.

Conclusions

A rational (with decreased weight, moisture-caused shape instability, and heat conductivity) macrostructure of plywood composite and its optimization methodology are proposed.

Rational geometrical properties of the proposed structure are obtained. Rational thickness of the top flange, bottom flange and ribs are 14, 13, and 30 mm, respectively, and total number of ribs is 27 in case if length of the plate is 2 m, width 1.25 m, thickness 4 cm, and load 2500 kg.

The obtained results are used to train Feed-Forward ANN consisting of 20 neurons in hidden and output layer and four input and output parameters. The validation test shows that the ANN could calculate rational parameters with good accuracy.

The architecture of ANN should be optimized for this kind of problems in the future.

Acknowledgements

This work has been supported by the European Social Fund within the project "Support for the implementation of doctoral studies at Riga Technical University".

References

Das, S. K.; Basudhar, P. K. 2006. Undrained lateral load capacity of piles in clay using artificial neural network, *Computers and Geotechnics* 33(8): 454–459. http://dx.doi.org/10.1016/j.compgeo.2006.08.006

European Committee for Standardization, 2004. Eurocode 5: design of timber structures. Part 1-1: general-common rules and rules for buildings.

Ferentinou, M. D.; Sakellariou, M. G. 2007. Computational intelligence tools for the prediction of slope performance, *Computers and Geotechnics* 34(5): 362–384. http://dx.doi.org/10.1016/j.compgeo.2007.06.004

Finnish Forest Industries Federation. 2002. *Handbook of finnish plywood*. Finland: Kirjapaino Markprint Oy. 64 p.

Garson, G. D. 1991. Interpreting neural network connection weights, *Artificial Intelligence Expert* 6(4): 46–51.

Goh, A. T. C. 1995. Back-propagation neural networks for modeling complex systems, *Artificial Intelligence in Engineering* 9(3): 143–151. http://dx.doi.org/10.1016/0954-1810(94)00011-S

Goldberg, D. E. 1989. *Genetic algorithms in search: optimisation and machine learning*. USA: Addison-Wesley. 432 p.

Goremikins, V.; Serdjuks, D. 2010. Rational structure of trussed beam, in *Proc. of the 10th International Conference "Modern Building Materials, Structures and Techniques*", 2010, Vilnius, Lithuania, 613–618.

Hagan, M. T.; Demuth, H. B.; Beale, M. H. 1996. *Neural network design*. Boston, MA: PWS Publishing. 736 p.

Hagan, M. T.; Menhaj, M. B. 1994. Training feedforward networks with the Marquardt algorithm, *IEEE Transactions on Neural Networks* 5(6): 989–993. http://dx.doi.org/10.1109/72.329697

He, S.; Li, J. 2008. Modeling nonlinear elastic behavior of reinforced soil using artificial neural networks, *Applied Soft Computing* 9(3): 954–961. http://dx.doi.org/10.1016/j.asoc.2008.11.013

Hoła, J.; Schabowicz, K. 2005. Application of artificial neural networks to determine concrete compressive strength based on non-destructive tests, *Journal of Civil Engineering and Management* 11(1): 23–32.

Maier, H. R.; Dandy, G. C. 2001. Neural network based modelling of environmental variables: a systematic approach, *Mathematical and Computer Modelling* 33(6–7): 669–682. http://dx.doi.org/10.1016/S0895-7177(00)00271-5

Rao, S. S. 2009. *Engineering optimization. Theory and practice*. 4th ed. New Jersey: John Willey & Sons Inc. 813 p.

Sliseris, J.; Rocens, K. 2010. Behaviour of multilayer sheet with technological imperfection, in *Proc. of the 10th International Conference "Modern Building Materials, Structures and Techniques*", 2010, Vilnius, Lithuania, 793–798.

Sliseris, J.; Rocens, K. 2011. Rational structure of panel with curved plywood ribs, in *"International Conference on Building Science and Engineering" (ICBSE 2011)*, 2011, Venice, Italy. *World Academy of Science, Engineering and Technology* 76: 317–323.

Jānis ŠLISERIS. Researcher and doctoral student in the Department of Structural Engineering at the Riga Technical University, Latvia. His research interests include the modern construction optimization and analysis, nonlinear technological mechanic of composite structures.

Kārlis ROCĒNS. Professor of structural engineering and director of the Institute of Structural Engineering and Reconstruction at the Riga Technical University, Latvia. He is a full member of Latvian Academy of Sciences and participant from Latvia in COST activity C25 "Sustainability of construction: Integrated approach to life time structural engineering". His research interests include the modern structures, technological mechanic of wood and composite materials and structural material science.