

A HYBRID METHOD OF BACKPROPAGATION AND PARTICLE SWARM OPTIMIZATION FOR ENHANCING ACCURACY PERFORMANCE

ABSTRACT

Aims: Backpropagation is an algorithm for adjusting the weight of neural networks in the training stage. The performance of backpropagation has proven superior in optimizing the weight of neural networks; however, this method needs improvement in the initiation stage, where the random process creates local optimal solution. Applying an algorithm based on the global search is an alternative to solve the drawback of backpropagation. One global search method with superior performance is particle swarm optimization. In this research, we apply the hybridization of backpropagation and particle swarm optimization (BP-PSO) to overcome the problem of backpropagation.

Study design: Research Papers and Short Notes.

Place and Duration of Study: Department of Informatics, Faculty of Mathematics and Natural Sciences, Udayana University, between June 2022 and November 2022.

Methodology: The dataset used in this study is a handwriting image dataset of the mathematical symbol. There are 240 symbols consisting of 180 images for training and 60 for testing. The robustness of the PSO method in obtaining the optimum global solution is expected to help backpropagation out of local optimal solutions. The application of PSO is carried out at the initial weight initialization stage of the artificial neural network. The tuning parameters of the artificial neural network are the number of neurons in the hidden layer and the value of the learning rate. There are three combinations in the number of neurons in the hidden layer, namely 10, 20, and 30. Meanwhile, the learning rate values are five different combinations, namely 0.1 to 0.9, the minimum error value is 0.01, and the maximum number of epochs is 1000. We carry out five repetitions in each test scenario.

Results: The performance results showed that PSO has succeeded in optimizing backpropagation, where the accuracy of the BP-PSO is higher than BP without optimization. The accuracy of BP-PSO is 97.2%, while the BP is 94.4%. The optimal learning rate value and the optimal number of hidden layers are 0.1 and 30 neurons, respectively.

Conclusion: The performance results showed that PSO has succeeded in optimizing backpropagation, where the accuracy of the BP-PSO is higher than BP without optimization. The optimization process of weighting the artificial neural network as the initial weight for later retraining shows a higher average accuracy, while decreasing the average number of epochs does not optimize initial weight.

Keywords: neural network, backpropagation, particle swarm optimization, classification, swarm intelligent

1. INTRODUCTION

One of the methods in artificial intelligence that is inspired by the biological workings of the human neural system is the artificial neural network (ANN) method. ANN is developed as a generalization of mathematical models from the human or neurological understanding, just like humans whose brains always learn from the environment. They do that to manage

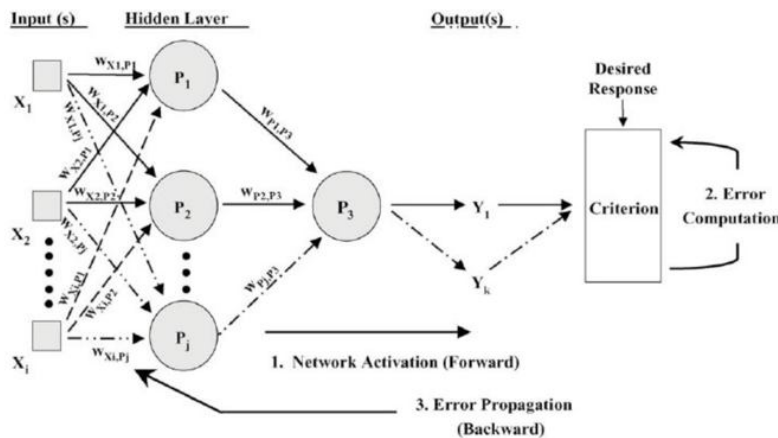
21 the environment adequately based on the gained experiences. ANN learning is performed
 22 iteratively as the network is presented with training examples, similar to how we learn from
 23 experience [7]. ANN has superior performance when compared to other methods [1]. This
 24 method requires a training process to be able to recognize class data. The training process
 25 is a stage of renewing the weight of the neural network through a series of learning iterations
 26 where the results of this training are the best weight values in recognizing patterns.
 27 Researchers have made various methods to optimize the weight value of this artificial neural
 28 network. One method that has superior performance is backpropagation [2]. The
 29 performance of backpropagation has been proven to have outstanding performance in
 30 optimizing the weight of the neural network. Although this algorithm is better than some
 31 others, it also has a weakness where the resulting results can be local optimal values [3].
 32 This weakness is influenced by the initial weight, which is randomly selected [4]. In some
 33 cases, backpropagation will experience poor performance compared to other method
 34 schemes. To solve this problem, we can insert a global search-based algorithm into
 35 backpropagation to find optimal initiation values of backpropagation weight.

36 There are various methods based on the search for global solutions, including
 37 genetic algorithms and multiple methods in intelligent swarms. If we compare the
 38 performance of genetic algorithms and swarm algorithms such as ant colony, bee colony,
 39 memetic, shuffled frog leap, and particle swarm optimization (PSO), the results show that the
 40 PSO algorithm has the most superior performance compared to these other algorithms [5].
 41 In integer programming problems, PSO has been compared with the Genetic algorithm. The
 42 comparison results show that the PSO algorithm is superior in terms of complexity,
 43 accuracy, iteration, and program simplicity in finding the optimal solution [6]. Seeing the
 44 excellent performance of this PSO, this research applies the PSO method to optimize the
 45 backpropagation method to get out of the local optimal solution. We evaluated the
 46 performance comparison between the backpropagation method with and without PSO at the
 47 testing stage. We also measured the effectiveness of the PSO in optimizing the weight of the
 48 training results.

50 2. METHODOLOGY

51 2.1. Backpropagation

52 Backpropagation is an algorithm for training artificial neural networks to obtain the
 53 optimal network weights. This set of weights determines the network's ability to recognize
 54 hidden patterns. Backpropagation uses an error output to readjust the weights in the
 55 backward direction. This study used a multi-layer backpropagation neural network scheme,
 56 as shown in Figure 1 [11].



57

58

Fig 1. Back-propagation Network

59 The followings are the steps of the Backpropagation algorithm.

- 60 1. Initialize the weights in the network with small random numbers.
- 61 2. A pattern of data training is fed into the network input layers.
- 62 3. Calculate all output unit in the hidden and output layers.

$$63 \quad z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (1)$$

$$64 \quad z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (2)$$

- 65 4. Calculate the unit error (δ) for every output layer and hidden layer. δ is the error unit
- 66 that will be used in updating the weight. t_k is the target output, Δw_{kj} is the weight
- 67 different, and α is the learning rate.

$$68 \quad \delta_k = (t_k - y_k)y_k(1 - y_k) \quad (3)$$

$$69 \quad \Delta w_{kj} = \alpha \delta_k z_j \quad (4)$$

- 70 5. Modify all the weights.

$$71 \quad w_{kj}(new) = w_{kj}(old) + \Delta w_{kj} \quad (5)$$

72 The subsequent pattern from the training set is chosen, and step number 2 will be carried
73 out until all patterns are processed. The algorithm ends when the error is less than the
74 minimum criteria.

75

76 2.2. Particle Swarm Optimization

77 Particle Swarm Optimization (PSO) is a population-based stochastic optimization
78 algorithm motivated by the intelligent collective behavior of animals such as flocks of birds or
79 schools of fish [13]. PSO, as an optimization tool, provides a population search-based
80 procedure. PSO performs searching via a swarm of particles that updates from iteration to
81 iteration. Each particle moves in the direction of its previous best (pBest) position and the
82 global best (gBest) position in the swarm [10] to obtain the optimal solution. The following
83 are the stages of the particle swarm optimization algorithm:

- 84 1. Initialize particles with random speed and velocity.
- 85 2. Evaluate the fitness value of each particle. Then the comparison process between
- 86 particles and Pbest is carried out. If the particle fitness value is better than the Pbest
- 87 value, the Pbest value is converted into the particle fitness value.
- 88 3. Compare the Gbest value against the best value for all particles. If the fitness value
- 89 obtained is better than the Gbest value, the Gbest value is converted into the particle
- 90 fitness value.
- 91 4. Perform the process of changing the speed and rank of particles using formulas (6) and
- 92 (7).
- 93 5. Repeat step 2 until it meets stopping criteria such as best fitness score or maximum
- 94 iteration.

95

96 Some of the popular terms in PSO are as follows:

- 97 ▪ Swarm: population/herd of an algorithm.
- 98 ▪ Particles: individuals in a swarm.
- 99 ▪ pBest (Personal best): the best rank of a particle.
- 100 ▪ gBest (Global best): the best position of the particle on a swarm.
- 101 ▪ Velocity (v): a vector that determines the movement of particles as they move.
- 102 ▪ Inertia weight (w): parameters controlling the speed of a particle.
- 103 ▪ Learning speed (c_1 and c_2): constants for particles (c_1) and swarm (c_2).

104

105 In making changes to velocity, PSO has three parts, namely the social part, cognitive part
 106 and momentum part. These three parts are used in the velocity tracking process. The
 107 following is the PSO equation with the inertia weight [12].
 108

$$109 \quad v_{ij}(t + 1) = w * v_{ij}(t) + c_1 * rand() * (p_{ij}(t) - x_{ij}(t)) + c_2 * rand() * (g_{ij} - x_{ij}(t)) \quad (6)$$

$$110 \quad x_{ij}(t + 1) = x_{ij}(t) + v_{ij}(t) \quad (7)$$

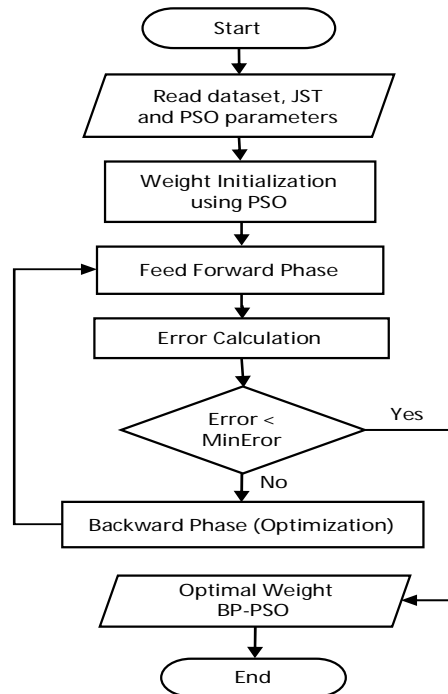
111
 112 Description :

- 113 v = Particel Velocity
- 114 x = Particel Position
- 115 w = Inertia Weight
- 116 c1 = pBest Constanta
- 117 c2 = gBest Constanta
- 118 p = pBest
- 119 g = gBest
- 120 rand() = Random Number

121
 122

123 2.3. Hybridization Scheme

124 The backpropagation learning algorithm's major drawback is local minimal problems [8]. The
 125 robustness of the PSO method in obtaining the optimum global solution is expected to help
 126 backpropagation out of the local optimal issues. The application of PSO is carried out at the
 127 initial weight initialization stage of the artificial neural network. The impact of random weight
 128 initialization can be seen in previous studies [9].
 129



130
 131
 132

Figure 2. Hybridization Scheme

133 3. RESULTS AND DISCUSSION

134 3.1. Hybridization

135 The dataset used in this study to measure the accuracy of PSO in optimizing
136 backpropagation is a mathematical symbol image. There are 20 math symbols written by
137 hand and using the image equation in the Microsoft Word Application. For each symbol, we
138 have 12 sample images. The total number of data is 240, and we divided it into two parts;
139 70% as training data and 30% as the test data. Before this dataset enters the pattern
140 recognition stage with the backpropagation and PSO methods, we first preprocessed the
141 data. The preprocessing steps are as follows:

- 142 1) Grayscale: In this process, the RGB image is converted into a grayscale or gray
143 image. The red, green, and blue component values will be taken for each image pixel.
144 The method used for grayscale is the weighted method. This method will produce
145 grayscale values by adding up 30% from red, 59% from green, and 11% from the blue.
- 146 2) Binarization: We convert the grayscale image into a black-and-white or binary image
147 using the thresholding method. The thresholding method uses a value to check the
148 possibility of whether the gray color of the image pixels is converted to black or white.
149 The threshold value used is 127 because it is in the middle of 0 to 255.
- 150 3) Segmentation: This binarization process uses a gray image where the color value of
151 each pixel will be taken and compared with the threshold value. If the color value is
152 greater than the threshold value (> 127), then it will be changed to 255 (white) and 0
153 (black color) otherwise. The result of this binary process is a binary image that only
154 consists of white and black.
- 155 4) Normalization: The image resulting from the segmentation process has different sizes,
156 so it is necessary to normalize the image size. There are two stages in the
157 normalization process, namely the padding and scaling stages. In the padding stage,
158 values will be added to the sides of an image that is not square. At the scaling stage,
159 the image size is changed to 64x64 pixels. First, we insert the image height and width
160 value and the image to be normalized. We then seek the largest value between the
161 height and width of the image. If the height value exceeds the width, the left and right
162 values will be added according to the difference between the height and the width
163 divided by 2. If the width value exceeds the height, the top and bottom values will be
164 added according to the difference between width and height divided by 2. Each side of
165 the image will be added according to the top, bottom, left, and right values so that the
166 image will have the same size. This stage is called the padding stage. We convert the
167 image size to 64x64 pixels in the scaling stage. The image produced from this
168 normalization process is a 64x64 pixels image.

169 After completing the four stages, we carry on with the feature extraction stage. This is done
170 to get the feature values for each dataset used in the pattern recognition stage. We extract
171 features from the image obtained from the preprocessing step in the feature extraction
172 stage. The method used to perform the feature extraction process is a gradient histogram.
173 This feature extraction method is based on the slope or direction of the pixels in relation to
174 the surrounding pixels. The result of this feature extraction is a vector which is a
175 characteristic of the image.

177 3.2. Training and Testing Result

178 This study's first result of system testing is the best artificial neural network architecture for
179 recognizing mathematical symbols. The parameters of the artificial neural network used are
180 the number of neurons in the hidden layer and the learning rate value. There are three
181 combinations in the number of neurons in the hidden layer, namely 10, 20, and 30.
182 Meanwhile, the learning rate values are five different combinations, namely 0.1 to 0.9. The
183 minimum error value is 0.01, and the maximum number of epochs is 1000. In each different
184 test scenario, we implemented five tests.

185 The second system test result is the optimization effect of the initial weight value of the
 186 neural network using the particle swarm optimization algorithm. The results of this study will
 187 compare the accuracy value of the backpropagation method neural network with a random
 188 initial weight (ANN BP) and the backpropagation neural network with the initial weight
 189 generated using PSO.

190 We investigated the system's accuracy in recognizing mathematical symbols in the testing
 191 stage. This study will make several changes to the learning rate to determine the effect on
 192 the resulting accuracy value. The learning rate value will be used to find the program's
 193 accuracy in recognizing mathematical symbols. The value of the accuracy rate is calculated
 194 using equation (8).

195
$$P(N) = \frac{IN}{N} * 100\% \quad (8)$$

196 Description :

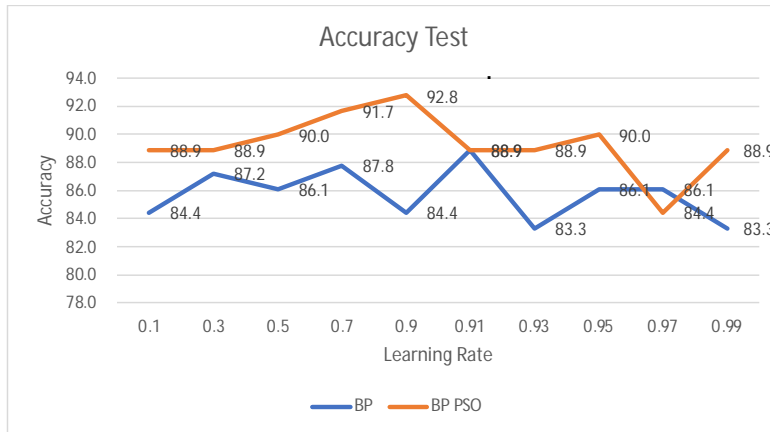
- 197 P (N) = level of accuracy
 198 IN = The number of successfully recognized data
 199 N = Total number of data

200
 201 Tables 1-3 are the test results with Hidden Neurons 10, 20, and 30. Figures 3-5 present the
 202 results of the test scenario in graphical form. From the results of the scenario testing that has
 203 been carried out, the optimal results of the backpropagation weight parameter are in the
 204 learning rate value. Figure 3 shows the changes in the performance of the backpropagation
 205 neural network with ten hidden neurons. The highest accuracy value of 92.8% is achieved
 206 when the learning rate is 0.9 for backpropagation with PSO, compared to 88.9% when the
 207 learning rate is 0.91 for backpropagation without PSO optimization. We can also see from
 208 the figure that in almost all tested learning rate values, BP PSO performance always has
 209 superior accuracy rates compared to the backpropagation algorithm without PSO
 210 optimization. This clearly shows the effectiveness of PSO in optimizing the backpropagation
 211 neural network.
 212
 213

Table 1. Test Results with 10 Hidden Neurons

Learning Rate	Epoch		Accuration	
	BP	BP PSO	BP	BP PSO
0.1	1000	996	84.4	88.9
0.3	398.2	342.4	87.2	88.9
0.5	295.4	218.4	86.1	90.0
0.7	329.2	150.6	87.8	91.7
0.9	146	117.4	84.4	92.8
0.91	321.4	116.2	88.9	88.9
0.93	319.4	121.2	83.3	88.9
0.95	295.4	218.4	86.1	90.0
0.97	125.4	107.2	86.1	84.4
0.99	121.6	104	83.3	88.9

214
 215



216
217

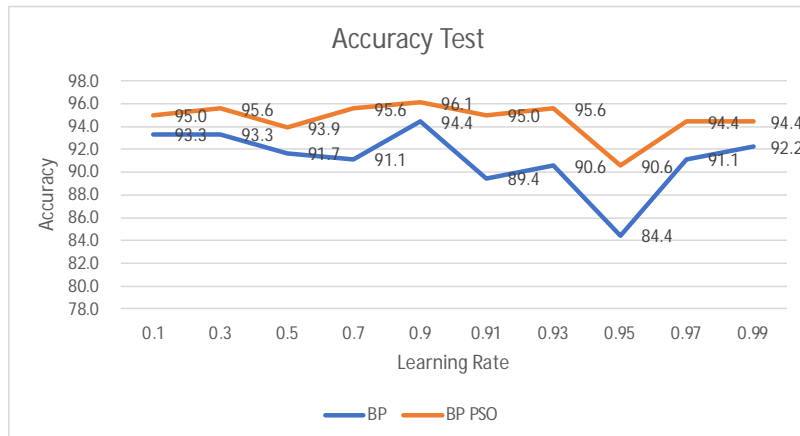
Fig 3. PSO Test Results with 10 Hidden Neurons

218

Table 2. Test Results with 20 Hidden Neurons

Learning Rate	Epoch		Accuracy	
	BP	BP PSO	BP	BP PSO
0.1	564.4	560.8	93.3	95.0
0.3	202	187.2	93.3	95.6
0.5	159.6	116.2	91.7	93.9
0.7	86.2	82.8	91.1	95.6
0.9	70.2	64.6	94.4	96.1
0.91	70.8	64.2	89.4	95.0
0.93	77.6	63.4	90.6	95.6
0.95	188	112.4	84.4	90.6
0.97	64.2	61.6	91.1	94.4
0.99	259.6	59.2	92.2	94.4

219
220



221
222

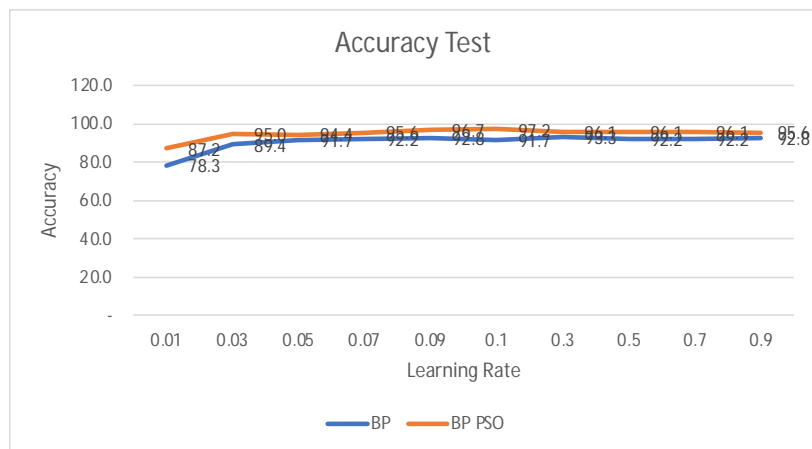
Fig 4. PSO Test Results with 20 Hidden Neurons

223

Table 3. Test Results with Hidden Neurons 30

224
225

Learning Rate	Epoch		Accuration	
	BP	BP PSO	BP	BP PSO
0.01	1000	1000	78.3	87.2
0.03	70.8	64.2	89.4	95.0
0.05	920.2	895.6	91.7	94.4
0.07	618.8	623.8	92.2	95.6
0.09	596.4	487.8	92.8	96.7
0.1	579.8	444	91.7	97.2
0.3	161.8	150.6	93.3	96.1
0.5	96.2	91.6	92.2	96.1
0.7	103.2	65.8	92.2	96.1
0.9	54.4	52.2	92.8	95.6



226
227

Fig 5. PSO Test Results with 30 Hidden Neurons

228 4. CONCLUSION

229 From the results of trials and evaluations of the research that has been done, the following
230 conclusions can be drawn.

- 231 1. We achieve the highest average accuracy of 94.4% with a 0.1 learning rate and 30
232 neurons in the hidden layer for neural network without optimization, and 97.2% otherwise.
- 233 2. Learning rate affects the neural network training process. A low learning rate of 0.1 will
234 lead to a better average accuracy than a high learning rate of 0.9. However, a contrasting
235 effect can be seen in the number of epochs. A more significant number of epochs will
236 lead to a better average accuracy when we use the same number of neurons in the
237 hidden layer.

238

239 ACKNOWLEDGEMENTS

240 We are extremely grateful to the Faculty of Mathematics and Natural Sciences, Udayana
241 University for the funding of this study. This study was funded under the Study Program's
242 Flagship research scheme. All the research activities were conducted at the Computing
243 Laboratory at the Informatics Department, Faculty of Mathematics and Natural Sciences,
244 Udayana University.

245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292

REFERENCES

1. Mahdiyah, U., Irawan, M.I., Imah, EM. Study Comparison Backpropagation, Support Vector Machine, and Extreme Learning Machine For Bioinformatics Data. *Journal of Computer Science and Information*. 2015;8(1): 53-59.
2. Amalia, O. A., Putri, A. H. M. Comparison of backpropagation artificial neural network and SARIMA in predicting the number of railway passengers. *Journal of Physics: Conference Series*. 2020;1663(2020):1-14.
3. Du, W. Research on Evaluation of Enterprise Performance Based on BP Neural Network Improved by Levenberg-Marquardt Algorithm. *Proceedings of the 2015 International Conference on Automation, Mechanical Control and Computational Engineering*. 24-26 April 2015, Jinan, Shandong. 2015;169-173.
4. Lee, A., Geem, Z.W., Suh, K.D. Determination of Optimal Initial Weights of An Artificial Neural Network by Using Harmony Search Algorithm. *Applied Sciences*. 2016;6(6):1-17.
5. Garg, A., Juneja, D. A Comparison and Analysis of various extended Techniques of Query Optimization. *International Journal of Advancements in Technology*. 2012;3(3):184-194.
6. Wihartiko, F.D., Wijayanti, H., Virgantari, F. Performance comparison of genetic algorithms and particle swarm optimization for model integer programming bus timetabling problem. *IOP Conference Series: Materials Science and Engineering*. 12 October 2017, Tangerang, Indonesia. 2017;1-6.
7. Basheera, I.A., Hajmeer, M. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*. 2000;43(1):3-31.
8. Burse, K., Manoria, M., Kirar, V. P. S. Improved Back Propagation Algorithm to Avoid Local Minima in Multiplicative Neuron Model. *International Journal of Electrical and Computer Engineering*. 2010;4(12):1776-1779.
9. Chang, Y.T., Lin, J., Shieh, J.S., Abbod, M.F. Optimization the Initial Weights of Artificial Neural Networks via Genetic Algorithm Applied to Hip Bone Fracture Prediction. *Hindawi Publishing Corporation Advances in Fuzzy Systems* 2012;2012:1-9.
10. Zhang, Y., Balochian, S., Agarwal, P. Bhatnagar, V., Housheya, O. J. Artificial intelligence and its applications. *Mathematical Problems in Engineering*. 2016;2(6):1-12.
11. Kaur, J., Kaur, P. A Review: Artificial Neural Network. *International Journal of Current Engineering and Technology*. 2018;8(4):904-910.
12. Kong, F., Jiang J., Huang, Y. An Adaptive Multi-Swarm Competition Particle Swarm Optimizer for Large-Scale Optimization. *Mathematics*. 2019;7(521):1-13.
13. Wang, D., Tan, D., Liu, L. Particle swarm optimization algorithm: an overview. *Springer-Verlag Berlin Heidelberg*. (2018);22(2):387-408