

Scenario-based wind speed estimation using a new hybrid metaheuristic model: Particle swarm optimization and radial movement optimization

Measurement and Control
2019, Vol. 52(5-6) 493–508
© The Author(s) 2019
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0020294019842597
journals.sagepub.com/home/mac
 SAGE

Alper Kerem¹ and Ali Saygin²

Abstract

This paper presents a new hybrid metaheuristic model in order to estimate wind speeds accurately. The study was started by the training process of artificial neural networks with some metaheuristic algorithms such as evolutionary strategy, genetic algorithm, ant colony optimization, probability-based incremental learning, particle swarm optimization, and radial movement optimization in the literature. The success of each model is recorded in graphs. In order to make the closest estimation and to increase the system stability, a new hybrid metaheuristic model was developed using particle swarm optimization and radial movement optimization, and the training process of artificial neural networks was performed with this new model. The data were obtained by real-time measurements from a 63-m-high wind measurement station built at the coordinates of UTM E 263254 and N 4173479, altitude 1313 m. Two different scenarios were created using actual data and applied to all models. It was observed that the error values in the designed new hybrid metaheuristic model were lower than those of the other models.

Keywords

Radial movement optimization, particle swarm optimization, artificial neural networks, wind speed estimation, hybrid metaheuristic algorithm

Date received: 17 January 2019; accepted: 12 March 2019

Introduction

Wind energy is an environment-friendly, unlimited, most economical type of energy which has a broad area of usage and is developing rapidly.^{1,2} However, the stochastic and intermittent structure of wind is the main factor that makes it difficult to estimate the speed and power of wind.^{3,4}

The demand for electricity production by wind energy is accelerating day by day. This demand brings about an increase in the number of wind farms as well as some related problems. The main ones among these problems may be listed as grid reliability, power system quality, and interconnected linking operations.^{2,5,6} Performing procedures of wind speed and wind power estimation is effective in solving such problems. In addition, the procedure of wind speed and wind power estimation is also highly important in terms of determining the locations of new wind farms to be built,⁷ facility maintenance and energy planning,⁸ energy transformation efficiency and reduction of surge risks,⁹ and optimal management of the benefits and risks of

wind farms.¹⁰ Wind speed and power estimation studies are discussed under four groups in terms of time. These are immediate-run estimation, short-run estimation, medium-run estimation, and long-run estimation.^{11–13}

Wind speed and wind power estimation studies may also be categorized based on their estimation models as the persistence method,^{13–15} physical method,¹⁶ statistical method,¹⁷ and hybrid method.¹⁸ The persistence method is the simplest method for estimating wind and is based on the assumption of a high correlation between the existent and future wind values. This

¹Department of Electrical-Electronics Engineering, Faculty of Engineering and Architecture, Kahramanmaraş Sutcu Imam University, Kahramanmaraş, Turkey

²Department of Electrical and Electronics Engineering, Faculty of Technology, Gazi University, Ankara, Turkey

Corresponding author:

Alper Kerem, Department of Electrical and Electronics Engineering, Faculty of Engineering and Architecture, Kahramanmaraş Sutcu Imam University, 46100 Kahramanmaraş, Turkey.
Email: alperkerem@ksu.edu.tr



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (<http://www.creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work without

further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

Table 1. Categorization of wind estimation in terms of time.¹³

Period	Interval	Practices
Immediate run	From a few minutes up to 30 min	Electricity market balancing Real-time grid operations Regulatory operations
Short run	From 30 min up to 6 h	Economic load distribution planning Making reasonable decisions on load Operational security in the electricity market
Medium run	From 6 h up to 1 day	Unit commitment decisions Reserve requirement decisions Generator online/offline status decisions
Long run	From 1 day up to 1 week or longer	Maintenance planning Establishment management Optimum operational cost Feasibility for wind farm design

method assumes that the wind speed value of any moment “ t ” will be the same as the value at a moment “ $t + \Delta t$.”^{13,15} In the physical method, wind speed estimations are based on physical parameters and meteorological projections. The statistical method involves wind power estimations to be produced using past wind data.^{19,20} Table 1 shows the categorization of wind estimation in terms of time.

While the physical method is highly successful in long-run estimations, the statistical method is successful in short-run estimations. The physical method uses parameters like terrain, slope, pressure, and temperature to make estimations of wind speed and power, while the statistical method uses statistical models.^{8,21,22} The hybrid method is a method that combines different methods that have unique abilities and characteristics. The purpose of such a method is to obtain the most suitable estimation performance by utilizing the advantages of each model that is included.¹⁸ Studies in the literature developed hybrid models which utilized two, three, and even four components. Some of these studies are listed in Table 2.

In this study, training studies of artificial neural networks (ANNs) were considered as a problem and it was aimed to develop a model to be used in the ANN training process which is accurate and fast and works stably. With this aim, the study was started using some metaheuristic algorithms in the literature in order to be used in the ANN training process. The success of the developed hybrid metaheuristic models of ANNs trained with evolutionary strategy (ES), ANNs trained with genetic algorithm (GA), ANNs trained with ant colony optimization (ACO), ANNs trained with probability-based incremental learning (PBIL), ANNs trained with particle swarm optimization (PSO), and ANNs trained with radial movement optimization (RMO) was recorded and plotted. In order to make the closest estimation and to increase the system stability, a new hybrid metaheuristic model was developed using PSO and RMO, and the training process of ANNs was performed with this new hybrid metaheuristic model. Two scenarios were created and all the models were applied.

The data were used in scenarios obtained from a wind measurement station built in a 63-m-high university campus. The success of the new designed hybrid metaheuristic model (ANNs trained with PRO + RMO) was compared with that of the other hybrid models.

Data processing

As shown in Figure 1, the wind measurement station was mounted at the coordinates of UTM E 263254 and N 4173479 and an altitude of 1313 m by steel guy ropes.

The three-dimensional (3D) depiction of the wind measurement station is shown in Figure 2.

The measurement mast consisted of 21 modules of 3 m height and a total height of 63 m. The devices that were installed on the wind measurement station are given in Table 3.

Installation works of the wind measurement station are shown in Figure 3.

Statistical values of the data set (August 2014) are given in Table 4.

Design of a new hybrid metaheuristic model

Metaheuristic algorithms are one step higher than heuristic algorithms, and they can select and apply heuristic algorithms based on the solution of the problem. They have some advantages of minimizing the complexity of calculations, success in nonlinear problems, capabilities of recording previous findings in local searches, and possibilities of producing effective results within a short time.^{45–47} The algorithms that constituted the model are presented in sections “RMO” and “PSO.”

RMO

RMO is a swarm-based, fast, simple, and effective metaheuristic optimization model which uses the spherical boundaries of a vector in the search space to find the optimal solution and was designed for the spherical optimization of complex and nonlinear optimization

Table 2. Hybrid models for wind speed estimation in the literature.

Hybrid models	Recording frequency	Number of records	Error models	References
GA + BP + EMD	10 min 1 h	700 500	MAPE, RMSE	Wang et al. ²³
FEEMD + CS + WNN	10 min	2930	MAE, MAPE, MSE	Xu et al. ²⁴
ANN + GFS	30 min	7856	MAE	Sanz et al. ²⁵
W + SVM	1-2-3-4-5 h	80	MAE, MAPE, RMSE	Chen et al. ²⁶
ARIMA + DENFIS	2 h 6 h 12 h	732 120 62	MAE, MAPE, RMSE	Ye et al. ²⁷
HHT + ANN	4 h	900	MAE, RMSE	Spiryaev et al. ²⁸
HHT + SVM	6 h	600		
HHT + RF	24 h	150		
HHT + BT				
AA + WLR	24 h	300	ARE	Guo et al. ²⁹
EMD + MkRVR	24 h	200	MAPE	Fei ³⁰
AR + GPR	1 h	1464	MAE, MAPE, RMSE	Zhang et al. ³¹
EWT + CSA + LSSVM	30 min	100	MAE, MAPE, RMSE	Hu et al. ³²
GA + APSO + WNN	10 min	3030	MAE, MAPE, MSE	Wang et al. ³³
EEMD + SVM	1 month	72	MAE, MAPE	Hu et al. ²⁰
EMD + FS + ANN/(DSF-ANN)	1 h	744	MAE, RMSE	Zhang et al. ³⁴
EMD + FS + SVM/(DSF-SVM)				
ANN + MC	a few minutes	4200	MAPE	Pourmousavi Kani and Ardehali ³⁵
ARIMA + ELM + SVM + LSSVM	15 min 30 min 30 min	576 700	MAPE MAE, MAPE, MSE	Wang and Hu ³⁶ Liu et al. ³⁷
FEEMD + MLP				
FEEMD + ANFIS				
WP + MLP				
WP + ANFIS				
SC + ESN + GA	15 min	5800	MAPE, MAE, nRMSE	Liu et al. ³⁸
ARIMA + ANN	1 h	48	ME, MAE, MSE	Cadenas and Rivera ³⁹
EMD + ANN	1 h	500	MAE, MSE, MAPE	Liu et al. ⁴⁰
ARIMA + KF + PSO	24 h	120	ARE, MAE, RMSE	Su et al. ⁴¹
WD + ABC + RVM	24 h	365	MAPE	Fei and He ⁴²
WP + BFGS	30 min	700	MAE, MAPE, MSE	Liu et al. ⁴³
WP + ARIMA + BFGS				
W + BFGS				
WT + SVM + GA	30 min	1440	MAE, MAPE, RMSE	Liu et al. ⁴⁴

ABC: artificial bee colony algorithm; ANFIS: adaptive neuro-fuzzy inference system; ANNs: artificial neural networks; ARIMA: auto-regressive integrated moving average; BP: backpropagation; ELM: extreme learning machine; EMD: empirical mode decomposition; ESN: echo state networks; FEEMD: fast ensemble empirical mode decomposition; FIS: fuzzy inference system; GA: genetic algorithm; GS: grid search; KF: Kalman filter; MC: Markov chain; PSO: particle swarm optimization; SC: spectral clustering; SVM: support vector machine; WD: wavelet decomposition; WT: wavelet transform; DENFIS: dynamic evolving neural-fuzzy inference system; GPR: Gaussian process regression; EWT: empirical wavelet transform; CSA: coupled simulated annealing; LSSVM: least-squares support vector machine; APSO: adaptive particle swarm optimization; WP: wavelet packet; MLP: multilayer perceptron; RVM: relevance vector machine; BFGS: Broyden–Fletcher–Goldfarb–Shanno; MAPE: mean absolute percentage error; MSE: mean square error; RMSE: root mean square error; MAE: mean absolute error; nRMSE: normalized root mean square error; WNN: Wavelet Neural Network; CS: Cuckoo Search; ME: Mean Error; GFS: Global Forecasting System; HHT: Hilbert–Huang Transform; RF: Radial Function; BT: Boosting Trees; AA: Apriori Algorithm; WLR: Weighted Local-Region; DSF: Decomposition Selection Forecasting; MkRVR: Multiple-Kernel Relevance Vector Regression; ARE: Absolute Relative Error.

**Figure 1.** Geographical location of the wind measurement station.

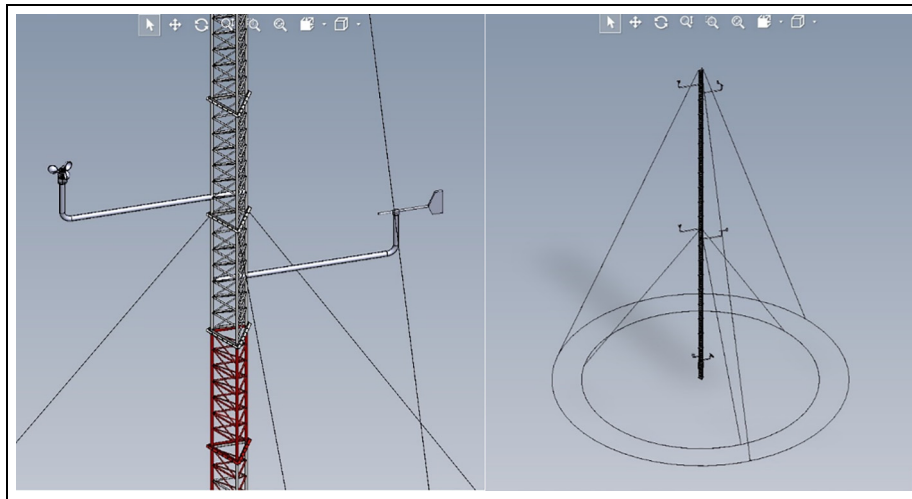


Figure 2. 3D depiction of the wind measurement station.



Figure 3. Installation works of the wind measurement station.

Table 3. Devices on the wind measurement station.

Devices	Height (m)
Anemometers	61
	30
Wind aspect sensors	59
	28
Temperature sensor	4
Humidity sensor	4
Pressure sensor	3.5
Data logger	3.5

problems.⁴⁶ In this algorithm, iteration cannot be transferred among the position and speed of the particles. Particles move from an updated point in each iteration, and therefore there is only a need for memory that has fewer calculations. The presence of a global best vector in the updating process prevents the algorithm from getting stuck in a local optimum.⁴⁸

The RMO algorithm is initiated by starting the particles in the search space that indicates the solutions of the problem. The positions of the particles are shown with a nop (number of particles) $\times$ nod (number of dimensions) matrix that is named as X_{ij} . Nop is normally equal to the number of variables that would be optimized, but it can be selected by the user on demand.^{48,49}

The step after obtaining the center point (cp) once is the dispersion of the particles from cp along the radius. This moves the particles from V_{ij} -vector-based cp along the radius as straight lines. The V_{ij} vector is a random $nop \times nop$ vector that is obtained by the equation below. The coefficient k in the equation should be a carefully selected integer^{45,47–49}

$$V_{ij} = rand(0, 1) \times V_{max(j)} \rightarrow \begin{cases} V_{max(j)} = \frac{X_{max(j)} - X_{min(j)}}{k} \\ i = 1, 2, \dots, nop \\ j = 1, 2, \dots, nod \end{cases} \quad (1)$$

As shown in equation (2), the speed vector has a weight of inertia (W) that is associated with and determines the convergence speed of the algorithm to an extent. This weight has a value that decreases along the generation iteration. Equation (2) shows the relationship between the weight of inertia and generation iterations^{45,47–49}

$$W_k = W_{max} - \left(\frac{W_{max} - W_{min}}{Generation_{max}} \right) \times Generation_k \quad (2)$$

As the speed vector $V_{i,j}^k$ is dependent on W_k , the values of W_{max} and W_{min} determine the effects of speed vectors on motion. These values can be selected by the user. The next step is to use the objective function to assess the suitability of all particles after dispersion^{45,47–49}

$$V_{i,j}^k = W_k \times rand(0, 1) \times V_{max(j)} \quad (3)$$

The particle that contains the optimal value is taken as the radial best ($Rbest$) vector. The accuracy value associated with the position of this particle represents the $Rbest$ particle. The positions of the $Gbest$ and $Rbest$ particles are used along with the updating vector (up) to update the optimal center point position as shown in equations (4) and (5). $C1$ and $C2$ are the coefficients that need to be adjusted before running the algorithm^{47–49}

$$cp^{k+1} = cp^k + up \quad (4)$$

$$up = C1 \times (Gbest - cp^k) + C2 \times (Rbest - cp^k) \quad (5)$$

Vectorial depiction of updating cp along the updating vector (up) is shown in Figure 4.

After the center point cp is updated, dispersion of the particles starts from the new cp . In the next generation, the $Gbest$ value in the previous generation is updated by comparison to the $Rbest$ value that is obtained during the dispersion of the particles in the new generation iteration. If $Rbest$ provides a better solution than $Gbest$, their positions should be switched. The process continues until $Gbest$ reaches a certain value or the number of generations reaches a defined maximum value.^{47,48} Figure 5 shows the sprinkling of the particles along the radii where V_{max} is the radius of the sphere and updating the cp by up vector.

PSO

PSO is a two-dimensional simulation of the collective behaviors of swarms of birds and fish in cases such as looking for food. It is a metaheuristic algorithm. It is efficient and effective in terms of calculation and easy in terms of comprehension and implementation.^{50,51}

PSO depicts each bird or fish as a particle and communities of birds or fish as swarms. In the algorithm, first, the initial positions, speeds, and population are randomly created. Accuracy values are calculated by determining the initial values for each particle in the population. The position value of the particle that has

Table 4. Statistical values of the data set (August 2014).

Parameters	Average	Median	Standard deviation	Mode	Max	Min	Skewness	Kurtosis	Variance	Number of samples
Wind speed 61 m (m/s)	4.592	4.435	0.991	5.165	8.797	2.432	0.939	2.604	0.981	4464
Wind speed 31 m (m/s)	4.313	4.153	0.927	4.258	8.431	2.343	1.129	3.283	0.859	4464
Wind aspect 59 m (°)	161.689	156.717	56.931	149.179	282.038	41.013	0.092	-0.729	3241.174	4464
Wind aspect 29 m (°)	175.019	168.025	59.581	142.464	299.036	29.847	0.035	-0.734	3549.893	4464
Temperature (°C)	23.101	23.396	2.356	21.087	28.409	17.217	-0.259	-0.436	5.551	4464
Humidity (%)	46.638	47.642	13.329	61.670	76.673	26.791	0.292	-1.089	177.654	4464
Pressure (mbar)	865.763	866.348	2.354	867.423	870.448	861.358	-0.136	-0.640	5.544	4464

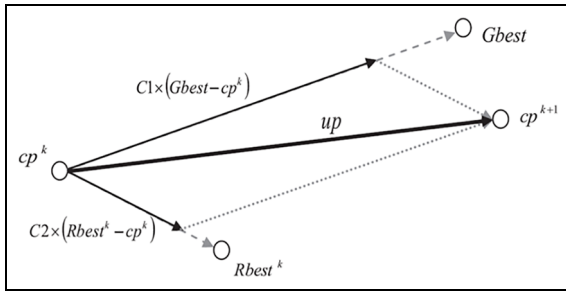


Figure 4. Vectorial depiction of updating cp along the updating vector (up).^{47,48}

the best (minimum value) among the accuracy values is selected as $Gbest$. The other particles update their positions and speeds based on this value. The cycle continues until the finalization step.⁵²

In PSO, X_i^k represents the position vector. The position vector in the problem space that consists of j particles is expressed as $X_i^k = [x_{i1}^k, x_{i2}^k, \dots, x_{ij}^k]$. V_i^k is the speed vector and the speed vector for a particle is shown as $V_i^k = [v_{i1}^k, v_{i2}^k, \dots, v_{ij}^k]$. Local best neighbors are the position vectors that have the local best values, and they consist of elements equal to the number of particles in each iteration. They are shown as $Pbest_i = [p_{i1}, p_{i2}, \dots, p_{ij}]$. Global best neighbors are the position values that have the best values in each iteration and are expressed as $Gbest = [g_1, g_2, \dots, g_j]$. The expression $c1$ is the effect of the best results obtained by the particle on the next iteration. The expression $c2$ is the effect of the best results obtained by the swarm on the next iteration. The expression w is the weight of inertia and it controls the effects of the previous speed vector on the current speed vector. In the case that this component which takes value in the range of 0–1 has a high value, the particle performs a search in the global environment, speeds up the solution, and may lead to overlooking the target area. In the case of low values,

the particle performs a search in the local environment and slows down the solution. The speeds and positions of the particles can be updated by equations (6) and (7)^{50–53}

$$v_i^{k+1} = v_i^k + c_1 \cdot rand_1^k (pbest_i^k - x_i^k) + c_2 \cdot rand_2^k (gbest^k - x_i^k) \quad (6)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (7)$$

Figure 6 shows the vectorial depiction of the motion of a particle k .

A new hybrid metaheuristic model

The main aim here is to design a new hybrid metaheuristic model that is different to those in the literature, has high accuracy rates, is fast, and works stably. Thus, the model was developed using PSO and RMO, and the training process of ANNs was performed with this new model. With this training process, it is aimed to optimize the linear and nonlinear components of the data sets and large random fluctuations. The flowchart of the new model is shown in Figure 7.

This new hybrid metaheuristic model was designed to make these two different features in a hybrid algorithm by taking advantage of PSO and RMO's distinctive individual features and to increase the system stability. Thus, PSO was used thanks to its features such as storage of coordinates and speeds of particles, ability to take into account the best available availability of particles, in determining the next movements, and it has benefited from its best past coordinates and its ability to consider the experiences of its most successful neighbor. Furthermore, RMO was used thanks to its excellent features such as the ability to search around the target point with an intensely focused feature, need for low memory, fast operation, and the

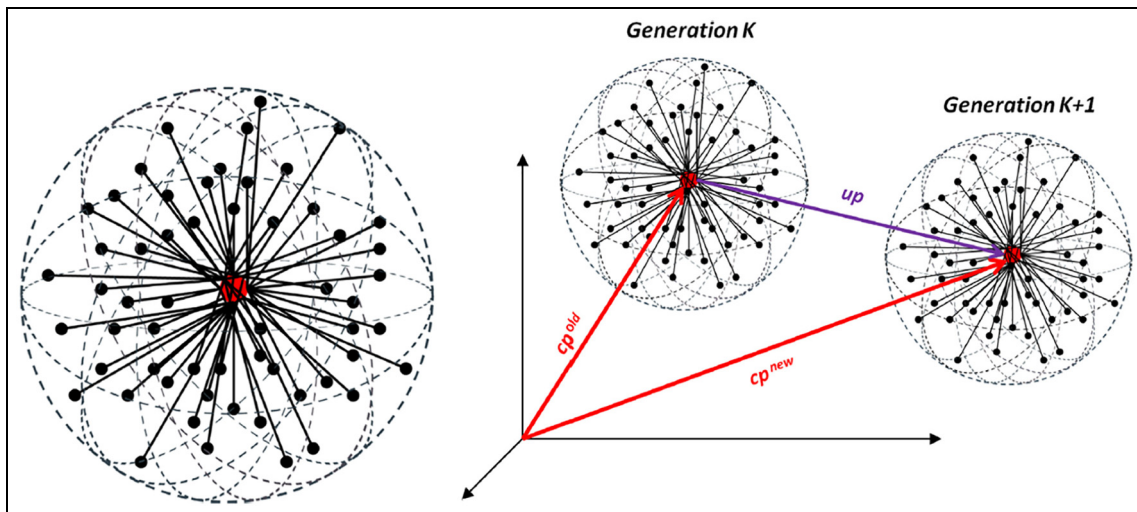


Figure 5. The sprinkling of the particles along the radii where V_{max} is the radius of the sphere and updating the cp by up vector.^{47–49}

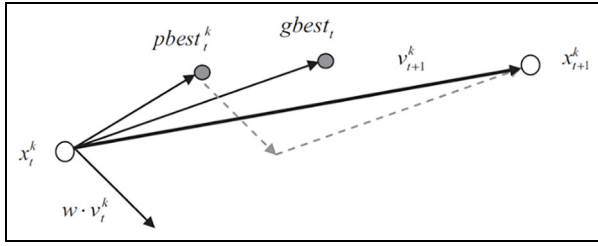


Figure 6. Vectorial depiction of the motion of a particle k .⁷

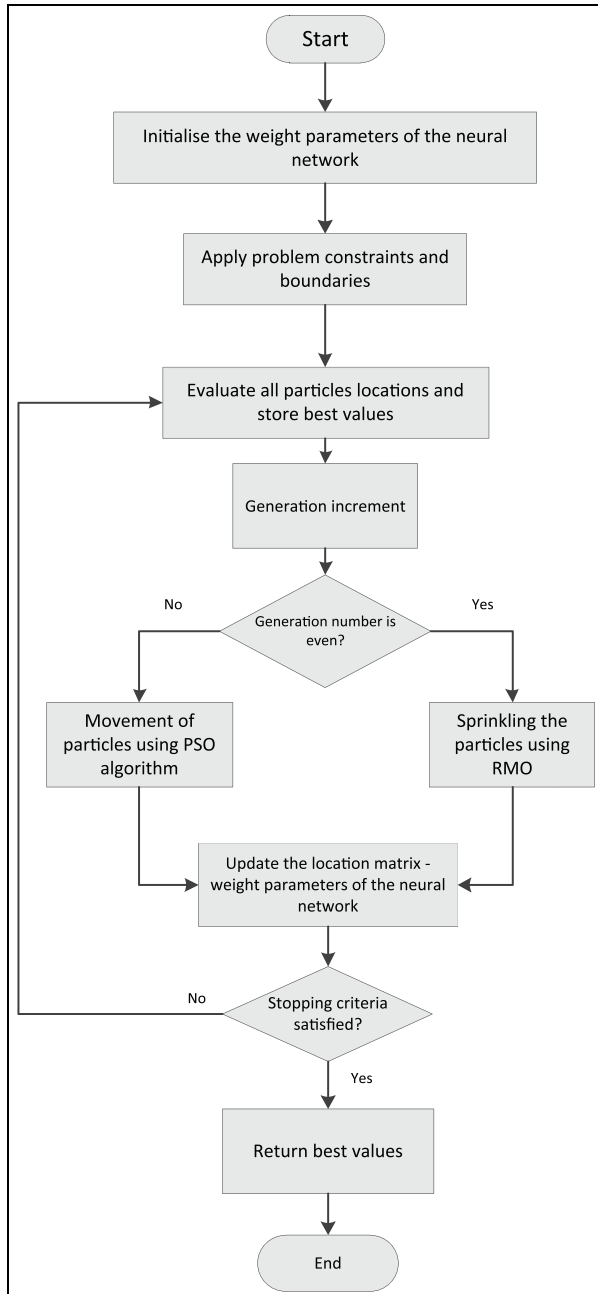


Figure 7. Flowchart of the new hybrid metaheuristic model.

ability to continue searching without getting pinched by the local optimum.

The data set was divided into three sets of data: 70% training, 15% verification, and 15% test data, while the model code is shown in Figure 8.

Interface for new hybrid metaheuristic model (ANNs trained with PSO + RMO)

A user interface was designed using MATLAB App Designer software. Thus, a new and an alternative hybrid metaheuristic estimation model that is highly accurate, fast working, practical, and easy to use was supported by an interface. The characteristics of the designed interface are given below:

- Allows selection of the file to be processed;
- Displays information about the data set being processed;
- Provides opportunity to use seven different estimation models (ANNs trained with ES, ANNs trained with GA, ANNs trained with ACO, ANNs trained with PBIL, ANNs trained with PSO, ANNs trained with RMO, and ANNs trained with PSO + RMO);
- Allows drawing real-time plots of the actual and estimated values;
- Can calculate six different error values such as normalized root mean square error (nRMSE), root mean square error (RMSE), mean square error (MSE), mean absolute percentage error (MAPE), peak signal-to-noise ratio (PSNR), and R and print them on the screen;
- Allows easy modification of the W_{max} , W_{min} , $c1$, and $c2$ parameters for PSO + RMO, the W , $C1$, $C2$, and $C3$ parameters for RMO, and the W_{max} , W_{min} , $c1$, and $c2$ parameters for PSO;
- Allows changing the number of iterations and number of particles (nop);
- Allows plotting the $Gbest$ values;
- Can bring up real-time information about the stage of the process on the screen;
- Can reset the command history with the “clear” button.

The interface designed using MATLAB App Designer is shown in Figure 9. The process starts by selecting the file to be run in the interface screen. Then the “create neural network” button is clicked on to create a neural network. The parameters in the bottom right corner of the screen are changed for a desired model (ANNs trained with ES, ANNs trained with GA, ANNs trained with ACO, ANNs trained with PBIL, ANNs trained with PSO, ANNs trained with RMO, ANNs trained with PSO + RMO) manually, that is, the parameter values that are considered to be the most suitable for a selected model are typed in. At this stage, it is also possible to change the number of iterations. Then the button for the desired model is clicked on. The progress of the process can be followed

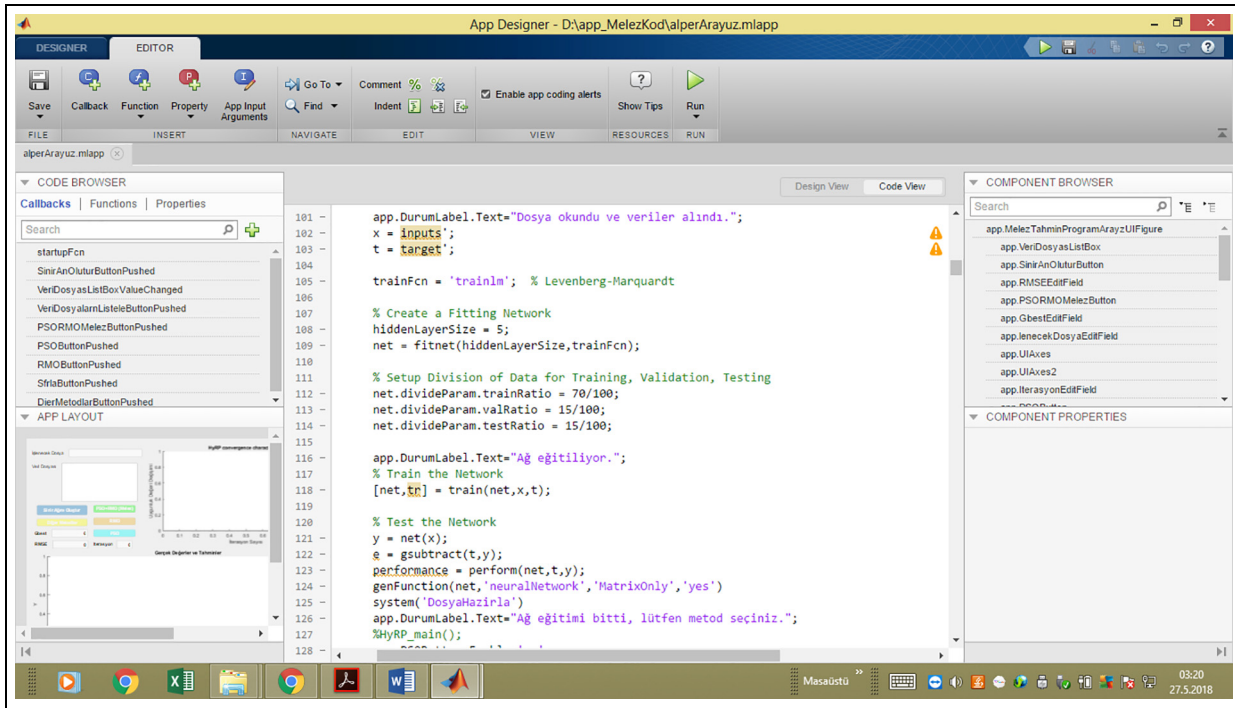


Figure 8. Dividing the data set into three parts.

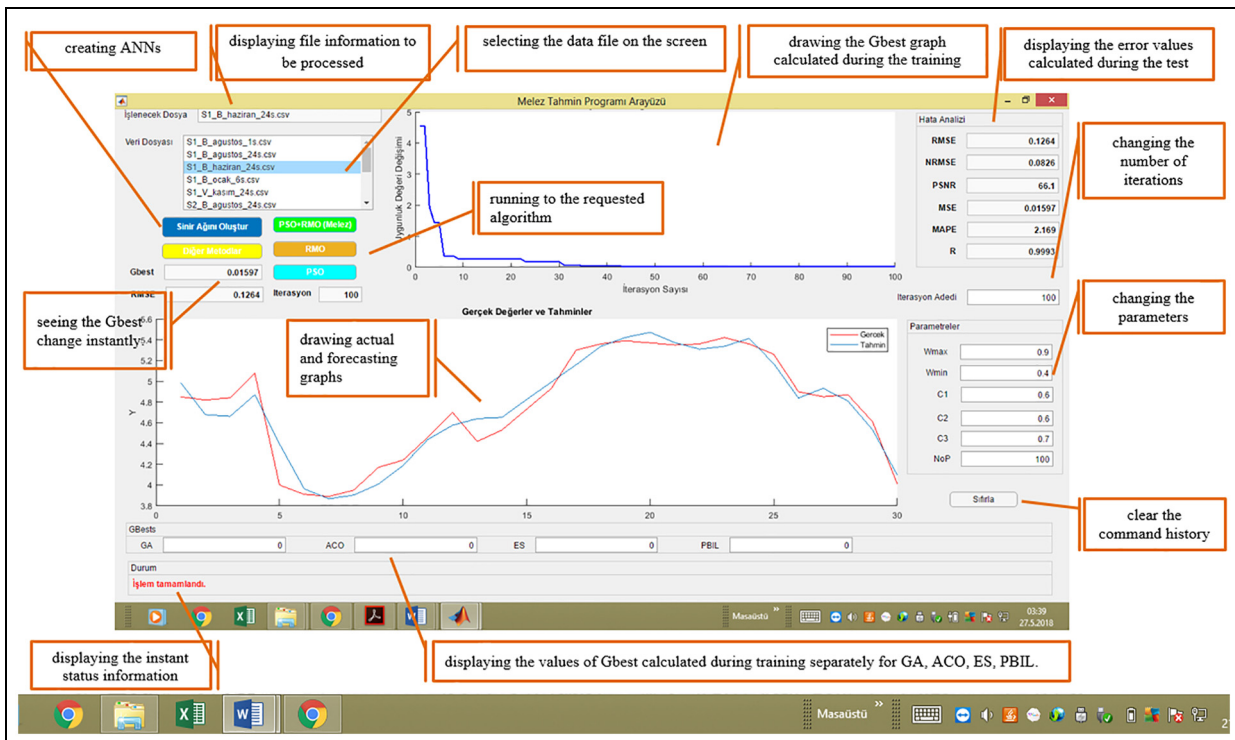


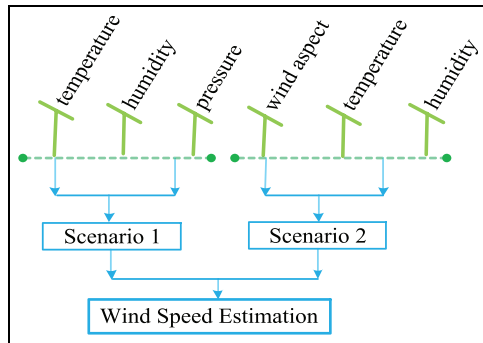
Figure 9. A user interface designed by MATLAB App Designer.

on the progress information part. After the process is completed, the actual and estimated values will be displayed on the screen along with the Gbest value. The error values (nRMSE, RMSE, MSE, MAPE, PSNR, R) are calculated one by one and shown on the error analysis part in the upper right corner of the screen.

Furthermore, it was seen that the ES, GA, ACO, and PBIL metaheuristic algorithms have been applied in various fields to solve a large number of problems and the results have been presented.⁵⁴⁻⁵⁶ These metaheuristic algorithms were used in this paper, thanks to excellent contributions of the literature.

Table 5. Input and output variables used in Scenario 1 and Scenario 2.

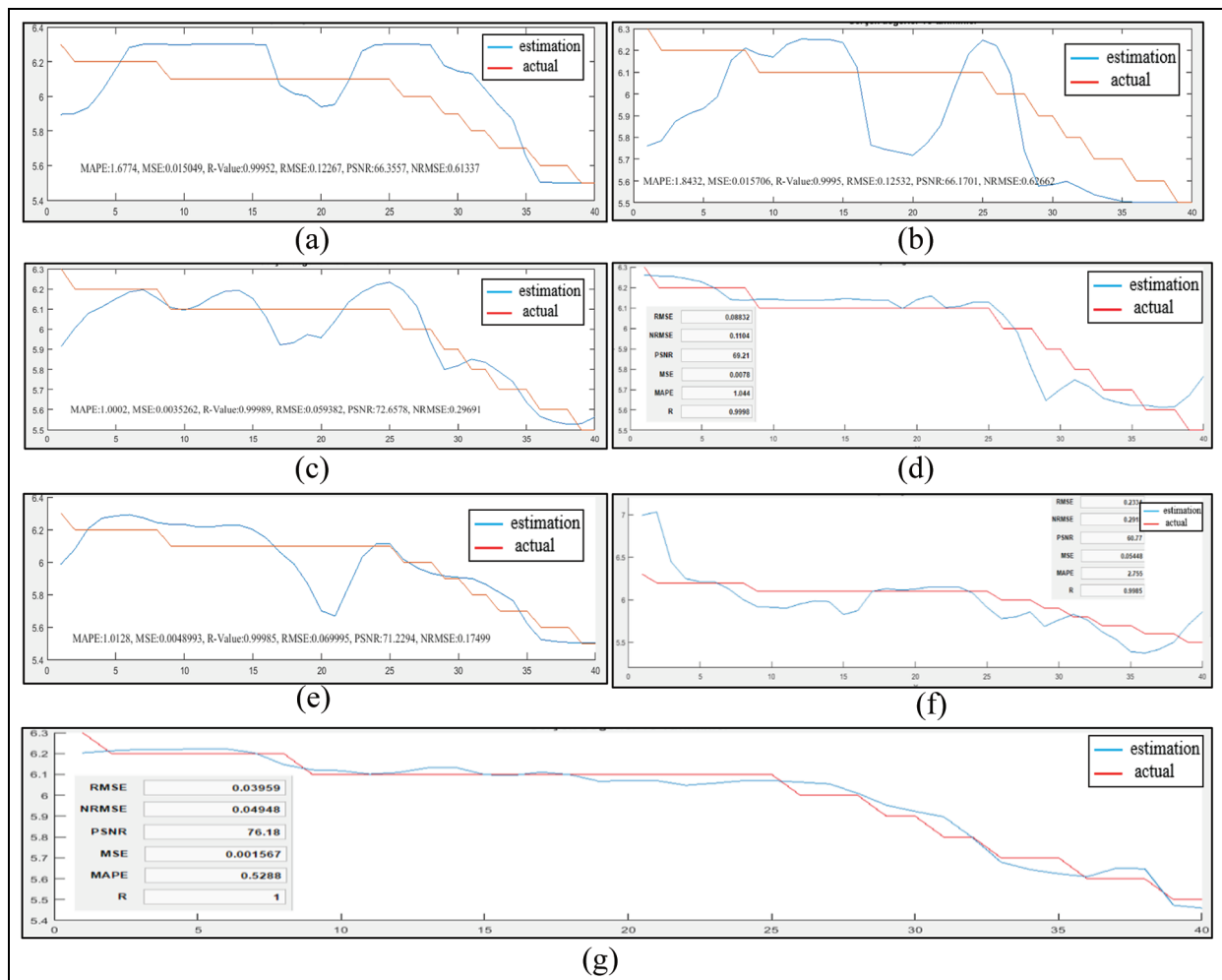
Variables and sensor heights used in estimation (input variables)				Estimated variables and sensor heights (output variables)
Wind aspect	Temperature	Humidity	Pressure	Wind speed
59 m	4 m	4 m	3.5 m	61 m

**Figure 10.** Scenario implementations.

Performance results

This section examines the performance of the designed hybrid metaheuristic estimation model for wind speed estimations. For this aim, the data sets obtained from the wind measurement station were converted into two different scenarios, and the performance results of the hybrid metaheuristic estimation model are presented. Figure 10 which shows the scenario implementations:

Scenario 1. In this scenario, wind speed estimations were made according to the data on temperature, humidity, and pressure. The data used were recorded in January 2014 (recording frequency: 6 h, recording time:

**Figure 11.** Actual and estimated plots of wind speed of Scenario I—January: (a) ANNs trained with ES, (b) ANNs trained with PBIL, (c) ANNs trained with GA, (d) ANNs trained with PSO, (e) ANNs trained with ACO, (f) ANNs trained with RMO, and (g) ANNs trained with PSO + RMO (new hybrid metaheuristic model).

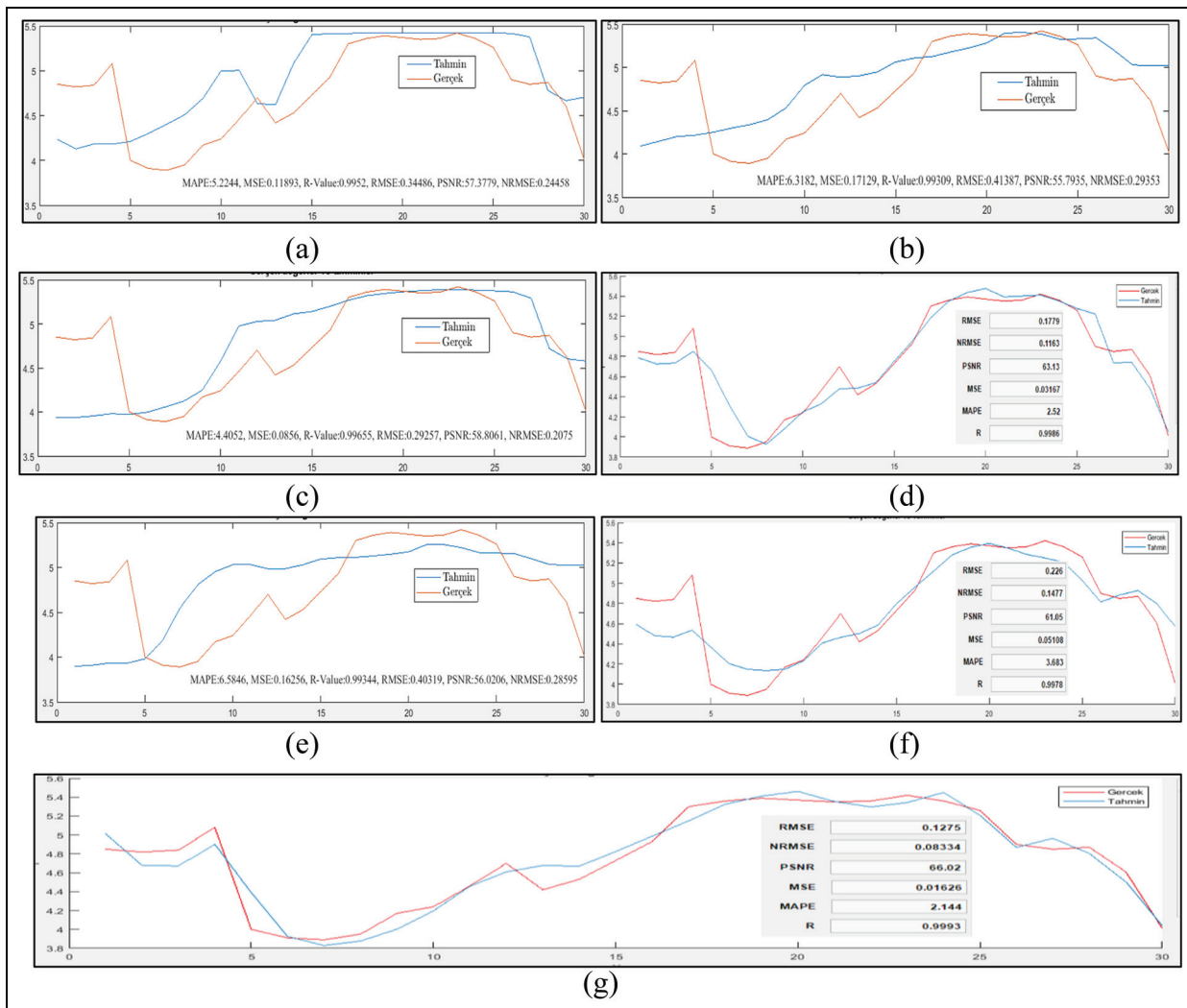


Figure 12. Actual and estimated plots of wind speed of Scenario 2—June: (a) ANNs trained with ES, (b) ANNs trained with PBIL, (c) ANNs trained with GA, (d) ANNs trained with PSO, (e) ANNs trained with ACO, (f) ANNs trained with RMO, and (g) ANNs trained with PSO + RMO (new hybrid metaheuristic model).

10 days) and June 2014 (recording frequency: 24 h, recording time: 30 days).

Scenario 2. In this scenario, wind speed estimations were made according to the data on wind aspect, temperature, and humidity. The data used were recorded in August 2014 (recording frequency: 24 h, recording time: 15 days). The input and output variables used in Scenario 1 and Scenario 2 are shown in Table 5.

Figure 11 shows the actual and estimated plots of wind speed of Scenario 1—January. It includes the success of wind speed estimation of the hybrid models ANNs trained with ES, ANNs trained with PBIL, ANNs trained with GA, ANNs trained with PSO, ANNs trained with ACO, ANNs trained with RMO, and ANNs trained with PSO + RMO.

Figure 12 shows the actual and estimated plots of wind speed of Scenario 1—June. It includes the success of wind speed estimation of the hybrid models ANNs trained with ES, ANNs trained with PBIL, ANNs trained with GA, ANNs trained with PSO, ANNs trained with ACO, ANNs trained with RMO, and ANNs trained with PSO + RMO.

trained with ACO, ANNs trained with RMO, and ANNs trained with PSO + RMO.

Figure 13 shows the actual and estimated plots of wind speed of Scenario 2—August. It includes the success of wind speed estimation of the hybrid models ANNs trained with ES, ANNs trained with PBIL, ANNs trained with GA, ANNs trained with PSO, ANNs trained with ACO, ANNs trained with RMO, and ANNs trained with PSO + RMO.

The best nRMSE, RMSE, MSE, MAPE, PSNR, and R values for the test error (50 runs) of Scenario 1—January are shown in Table 6. The min, average, max, and standard deviation values for Scenario 1—January test error (MAPE, 50 runs) are given in Table 7.

The best nRMSE, RMSE, MSE, MAPE, PSNR, and R values for the test error (50 runs) of Scenario 1—June are given in Table 8. The min, average, max, and standard deviation values for Scenario 1—June test error (MAPE, 50 runs) are given Table 9.

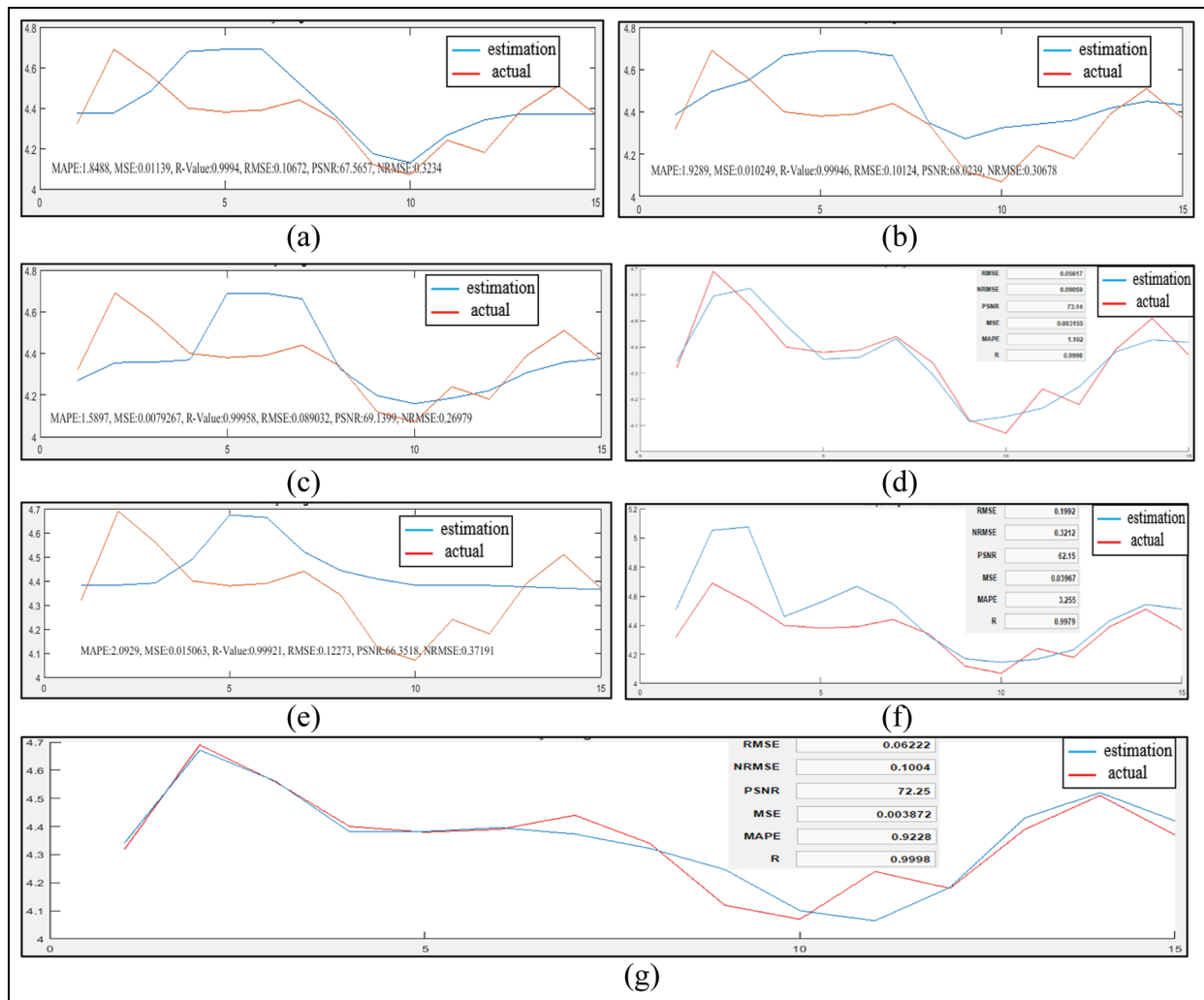


Figure 13. Actual and estimated plots of wind speed of Scenario 2—August: (a) ANNs trained with ES, (b) ANNs trained with PBIL, (c) ANNs trained with GA, (d) ANNs trained with PSO, (e) ANNs trained with ACO, (f) ANNs trained with RMO, and (g) ANNs trained with PSO + RMO (new hybrid metaheuristic model).

The best nRMSE, RMSE, MSE, MAPE, PSNR, and R values for the test error (50 runs) of Scenario 2—August are given in Table 10. The min, average, max, and standard deviation values for Scenario 2—August test error (MAPE, 50 runs) are given in Table 11.

Conclusion

Wind may show momentary changes due to its nature, and it causes highly difficult and complicated estimations. Wind speed and wind power estimation studies done as accurately as possible are highly important in terms of determining the locations for wind farms, shaping energy unit costs, efficient and effective energy investments, and grid reliability.

In this study, a new hybrid metaheuristic (ANNs trained with PSO + RMO) estimation model was designed using the PSO and RMO models. It means that the training process of ANNs was performed using PSO and RMO. The designed new hybrid metaheuristic model was compared to each of the other hybrid

models (ANNs trained with ES, ANNs trained with GA, ANNs trained with ACO, ANNs trained with PBIL, ANNs trained with PSO, and ANNs trained with RMO). An interface was designed using MATLAB App Designer (R2017b) and the whole model was visualized.

The performance of ANNs trained with ES, ANNs trained with GA, ANNs trained with ACO, ANNs trained with PBIL, ANNs trained with PSO, ANNs trained with RMO, and ANNs trained with the PSO + RMO combination were observed separately by two different scenarios (Scenario 1 and Scenario 2). According to the analysis of Scenario 1 (January and June) given in Tables 6 and 8, the nRMSE, RMSE, MSE, MAPE, PSNR, and R error values (test error, 50 runs) for the new hybrid metaheuristic model (ANNs trained with PSO + RMO) were much lower than those of the other hybrid models. Likewise, according to the analysis of Scenario 2 (August) given in Table 10, the nRMSE, RMSE, MSE, MAPE, PSNR, and R error values (test error, 50 runs) for the new hybrid

Table 6. The best nRMSE, RMSE, MSE, MAPE, PSNR, and R values for the test error (50 runs) of Scenario I—January.

Test error (50 runs) of Scenario I (recording frequency: 6 h, recording time: 10 days, January)											ANNs trained with ACO					ANNs trained with ES					ANNs trained with PBIL							
											nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R
											0.174	0.069	0.004	1.012	71.229	0.999	0.613	0.122	0.015	1.677	66.355	0.999	0.626	0.125	0.015	1.843	66.17	0.999
ANNs trained with GA											ANNs trained with PSO					ANNs trained with RMO					ANNs trained with PSO + RMO							
nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R					
0.296	0.059	0.003	1.000	72.657	0.999	0.110	0.088	0.007	1.044	69.210	0.999	0.291	0.233	0.054	2.755	60.77	0.998	0.049	0.039	0.001	0.528	76.18	1.000					

nRMSE: normalized root mean square error; RMSE: root mean square error; MSE: mean square error; MAPE: mean absolute percentage error; PSNR: peak signal-to-noise ratio; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

Table 7. The min, average, max, and standard deviation (SD) values for Scenario I—January test error (MAPE, 50 runs).

Test error (MAPE, 50 runs) of Scenario I (recording frequency: 6 h, recording time: 10 days, January)					ANNs trained with ACO					ANNs trained with ES					ANNs trained with PBIL				
					Min	Avg	Max	SD		Min	Avg	Max	SD		Min	Avg	Max	SD	
					1.012	4.881	9.929	2.655		1.677	5.962	10.033	2.540		1.843	4.107	8.616	2.109	
ANNs trained with GA					ANNs trained with PSO					ANNs trained with RMO					ANNs trained with PSO + RMO				
Min	Avg	Max	SD		Min	Avg	Max	SD		Min	Avg	Max	SD		Min	Avg	Max	SD	
1.000	4.751	9.016	2.491		1.044	1.387	3.734	0.642		2.755	6.827	10.07	1.957		0.528	0.752	2.142	0.413	

MAPE: mean absolute percentage error; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

Table 8. The best nRMSE, RMSE, MSE, MAPE, PSNR, and R values for the test error (50 runs) of Scenario I—June.

Test error (50 runs) of Scenario I (recording frequency: 24 h, recording time: 30 days, June)						ANNs trained with ACO						ANNs trained with ES						ANNs trained with PBIL					
						nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R
						0.285	0.403	0.162	6.584	56.020	0.993	0.244	0.244	0.118	5.224	57.377	0.995	0.293	0.413	0.171	6.318	55.793	0.993
ANNs trained with GA						ANNs trained with PSO						ANNs trained with RMO						ANNs trained with PSO + RMO					
nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R
0.207	0.292	0.085	4.405	58.806	0.996	0.116	0.177	0.031	2.520	63.130	0.998	0.147	0.226	0.051	3.683	61.050	0.997	0.083	0.127	0.016	2.144	66.020	0.999

nRMSE: normalized root mean square error; RMSE: root mean square error; MSE: mean square error; MAPE: mean absolute percentage error; PSNR: peak signal-to-noise ratio; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

Table 9. The min, average, max, and standard deviation (SD) values for Scenario 1—June test error (MAPE, 50 runs).

Test error (MAPE, 50 runs) of Scenario I (recording frequency: 24 h, recording time: 30 days, June)															
ANNs trained with ACO				ANNs trained with ES				ANNs trained with PBIL							
Min	Avg	Max	SD	Min	Avg	Max	SD	Min	Avg	Max	SD				
6.584	9.176	10.588	1.171	5.224	9.623	12.540	1.244	6.318	9.284	13.127	1.441				
ANNs trained with PSO				ANNs trained with RMO				ANNs trained with PSO + RMO							
Min	Avg	Max	SD	Min	Avg	Max	SD	Min	Avg	Max	SD				
2.520	3.856	7.516	1.234	3.683	17.974	27.320	7.288	2.144	2.158	2.372	0.055				

MAPE: mean absolute percentage error; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

Table 10. The best nRMSE, RMSE, MSE, MAPE, PSNR, R results for the test error (50 runs) of Scenario 2—August.

Test error (50 runs) of Scenario 2 (recording frequency: 24 h, recording time: 15 days, August)																	
ANNs trained with ACO						ANNs trained with ES						ANNs trained with PBIL					
nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R
0.371	0.122	0.015	2.092	66.351	0.999	0.323	0.106	0.011	1.849	67.565	0.999	0.306	0.101	0.010	1.929	68.023	0.999
ANNs trained with PSO						ANNs trained with RMO						ANNs trained with PSO + RMO					
nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R	nRMSE	RMSE	MSE	MAPE	PSNR	R
0.269	0.089	0.007	1.589	69.139	0.999	0.090	0.056	0.003	1.102	73.140	0.999	0.321	0.199	0.039	3.255	62.150	0.997
												0.100	0.062	0.003	0.922	72.250	0.999

nRMSE: normalized root mean square error; RMSE: root mean square error; MSE: mean square error; MAPE: mean absolute percentage error; PSNR: peak signal-to-noise ratio; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

Table 11. The min, average, max, and standard deviation (SD) values for Scenario 2—August test error (MAPE, 50 runs).

Test error (MAPE, 50 runs) of Scenario 2 (recording frequency: 24 h, recording time: 15 days, August)				ANNs trained with ACO				ANNs trained with ES				ANNs trained with PBIL			
				Min	Avg	Max	SD	Min	Avg	Max	SD	Min	Avg	Max	SD
				2.093	5.044	7.585	1.629	1.849	5.786	7.559	1.498	1.929	4.837	7.237	1.531
ANNs trained with GA				ANNs trained with PSO				ANNs trained with RMO				ANNs trained with PSO + RMO			
Min	Avg	Max	SD	Min	Avg	Max	SD	Min	Avg	Max	SD	Min	Avg	Max	SD
1.589	4.542	7.572	1.718	1.102	2.031	5.977	1.311	3.255	10.490	18.530	4.808	0.922	1.222	2.063	0.210

MAPE: mean absolute percentage error; ANN: artificial neural network; ACO: ant colony optimization; ES: evolutionary strategy; PBIL: probability-based incremental learning; GA: genetic algorithm; PSO: particle swarm optimization; RMO: radial movement optimization.

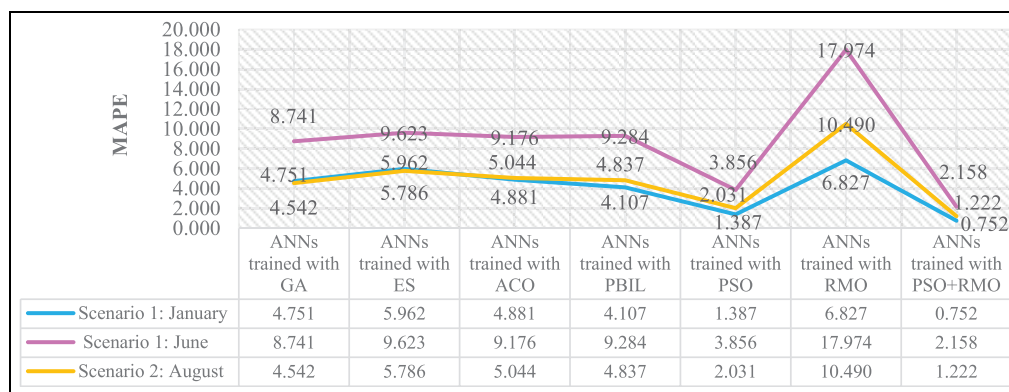


Figure 14. The average MAPE values for Scenario 1 and Scenario 2 (test error, 50 runs).

metaheuristic model (ANNs trained with PSO + RMO) were much lower than those of the other hybrid models. It shows the success of the new designed hybrid metaheuristic estimation model.

The average MAPE values for Scenario 1 and Scenario 2 (test error, 50 runs) are shown in Figure 14. A newly designed hybrid metaheuristic model (ANNs trained with PSO + RMO) has achieved the most successful results among all the hybrid models. These are the lowest errors among the all models, 0.752, 1.222, and 2.158 for Scenario 1 (January), Scenario 2 (August), and Scenario 1 (June), respectively.

As a result of the analyses, it was observed that ANNs trained with the PSO + RMO model showed a highly accurate and reliable performance in the solution of nonlinear problems such as wind speed estimation. Thus, a new hybrid metaheuristic model (ANNs trained with PSO + RMO) which has high accuracy, is fast, and works stably has become a contribution to the literature with this study.

Acknowledgements

The authors thank the project coordinator Prof. Dr Serdar Salman who provided insight and expertise that greatly assisted the research. Furthermore, special thanks to Drs Rasoul Rahmani and İsmail Kırbaş for their excellent contribution to this study.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

This research was supported by West Mediterranean Development Agency (BAKA, Project No. TR61/13/DFD/036) and Mehmet Akif Ersoy University Scientific Research Projects Commission (Project No. 0212-Gudumlu-13).

ORCID iD

Alper Kerem  <https://orcid.org/0000-0002-9055-5098>

References

1. Xiaochen W, Peng G and Xiaobin H. A review of wind power forecasting models. *Ener Procedia* 2011; 12: 770–778.
2. Yang M, Yun-Peng J, Qian W, et al. A combination method research for wind power prediction based on dynamic weight. *J Northeast Dianli Univ* 2013; 33(1–2): 131–136.
3. Kassa Y, Zhang JH, Zheng DH, et al. A GA-BP hybrid algorithm based ANN model for wind power prediction. In: *Proceedings of the 4th IEEE international conference on smart energy grid engineering*, Oshawa, ON, Canada, 21–24 August 2016.
4. Mao Y and Wang S. A review of wind power forecasting & prediction. In: *Proceedings of the international conference on probabilistic methods applied to power systems (PMAPS)*, Beijing, China, 16–20 October 2016.
5. Lund H. Large-scale integration of wind power into different energy systems. *Energy* 2005; 30(13): 2402–2412.
6. Debbağ Y and Yilmaz EN. Internet based monitoring and control of a wind turbine via PLC. In: *Proceedings of the 3rd international Istanbul smart grid congress and fair (ICSG)*, Istanbul, Turkey, 29–30 April 2015, pp. 1–5. New York: IEEE.
7. Varanasi J and Tripathi MM. A comparative study of wind power forecasting techniques: a review article. In: *Proceedings of the 3rd international conference on computing for sustainable global development (INDIACom)*, New Delhi, India, 16–18 March 2016.
8. Rasoul R, Yusof R, Seyedmehmoudian M, et al. Hybrid technique of ant colony and particle swarm optimization for short term wind energy forecasting. *J Wind Eng Ind Aerod* 2013; 123: 163–170.
9. Riah Y and Abedi M. Short term wind speed forecasting for wind turbine applications using linear prediction method. *Renew Energ* 2008; 33(1): 35–40.
10. Jing S, Jinmei G and Songtao Z. Evaluation of hybrid forecasting approaches for wind speed and power generation time series. *Renew Sustain Ener Rev* 2012; 16: 3471–3480.
11. Liu Y. *Machine learning for wind power prediction*. Master's Thesis, The University of New Brunswick, Saint John, NB, Canada, June 2016.
12. Kaur D. *Comparison of the effectivity of wind speed forecasting methods*. PhD Thesis, Auckland University of Technology, Auckland, New Zealand, 2015.

13. Soman SS, Zareipour H, Malik O, et al. A review of wind power and wind speed forecasting methods with different time horizons. In: *Proceedings of the 2010 North American power symposium*, Arlington, VA, 26–28 September 2010, pp. 1–8. New York: IEEE.
14. Chang WY. A literature review of wind forecasting methods. *J Power Ener Eng* 2014; 2: 161–168.
15. Wu YK and Hong JS. A literature review of wind forecasting technology in the world. In: *Power Tech 2007*, Lausanne, 1–5 July 2007, pp. 504–509. New York: IEEE.
16. Potter CW and Negnevitsky M. Very short-term wind forecasting for Tasmanian power generation. *IEEE Trans Power Syst* 2006; 21(2): 965–972.
17. Mao Y and Shaoshuai W. A review of wind power forecasting & prediction. In: *International conference on probabilistic methods applied to power systems (PMAPS)*, Beijing, China, 16–20 October 2016.
18. Bhaskar K and Singh SN. AWNN-assisted wind power forecasting using feed-forward neural network. *IEEE Trans Sustain Ener* 2012; 3: 306–315.
19. Ladislav Z. Wind speed forecast correction models using polynomial neural networks. *Renew Ener* 2015; 83: 998–1006.
20. Hu J, Wang J and Zeng G. A hybrid forecasting approach applied to wind speed time series. *Renew Energy* 2013; 60: 185–194.
21. Liu H, Tian H-Q, Chen C, et al. A hybrid statistical method to predict wind speed and wind power. *Renew Ener* 2010; 35(8): 1857–1861.
22. Abdoos AA. A new intelligent method based on combination of VMD and ELM for short term wind power forecasting. *Neurocomputing* 2016; 203: 111–120.
23. Wang S, Zhang N, Wu L, et al. Wind speed forecasting based on the hybrid ensemble empirical mode decomposition and GA-BP neural network method. *Renew Ener* 2016; 94: 629–636.
24. Xu X, Song X, Wang Q, et al. Hybrid forecasting model based data mining and cuckoo search: a case study of wind speed time series. *Open Fuels Ener Sci J* 2016; 9: 65–76.
25. Sanz SS, Pérez-Bellido Á, Ortiz-García E, et al. Short-term wind speed prediction by hybridizing global and mesoscale forecasting models with artificial neural networks. In: *Eighth international conference on hybrid intelligent systems*, Barcelona, 10–12 September 2008, pp. 608–612. New York: IEEE.
26. Chen N-Y, Qian Z and Meng X-F. A hybrid model for short-term wind speed forecasting based on wavelet and support vector machine. In: *IET conference on renewable power generation (RPG 2011)*, Edinburgh, 6–8 September 2011.
27. Ye R, Suganthan PN, Srikanth N, et al. A hybrid ARIMA-DENFIS method for wind speed forecasting. In: *IEEE international conference on fuzzy systems (FUZZ)*, Hyderabad, India, 7–10 July 2013.
28. Spiryaev V, Zhukov A and Leahy P. A hybrid wind speed forecasting strategy based on Hilbert-Huang transform and machine learning algorithms. In: *2014 international conference on power system technology (POWERCON 2014)*, Chengdu, China, 20–22 October 2014, pp. 2980–2986. New York: IEEE.
29. Guo Z, Chi D, Wu J, et al. A new wind speed forecasting strategy based on the chaotic time series modelling technique and the Apriori algorithm. *Ener Convers Manage* 2014; 84: 140–151.
30. Fei S-W. A hybrid model of EMD and multiple-kernel RVR algorithm for wind speed prediction. *Electr Power Ener Syst* 2016; 78: 910–915.
31. Zhang C, Wei H, Zhao X, et al. A Gaussian process regression based hybrid approach for short-term wind speed prediction. *Ener Convers Manage* 2016; 126: 1084–1092.
32. Hu J, Wang J and Ma K. A hybrid technique for short-term wind speed prediction. *Energy* 2015; 81: 563–574.
33. Wang J, Zhang F, Liu F, et al. Hybrid forecasting model-based data mining and genetic algorithm-adaptive particle swarm optimisation: a case study of wind speed time series. *IET Renew Power Generat* 2016; 10(3): 287–298.
34. Zhang C, Wei H, Zhao J, et al. Short-term wind speed forecasting using empirical mode decomposition and feature selection. *Renew Ener* 2016; 96: 727–737.
35. Pourmousavi Kani SA and Ardehali M. Very short-term wind speed prediction: a new artificial neural network-Markov chain model. *Energy Convers Manage* 2011; 52: 738–745.
36. Wang J and Hu J. A robust combination approach for short-term wind speed forecasting and analysis—combination of the ARIMA (Autoregressive Integrated Moving Average), ELM (Extreme Learning Machine), SVM (Support Vector Machine) and LSSVM (Least Square SVM) forecasts using a GPR (Gaussian Process Regression) model. *Energy* 2015; 93: 41–56.
37. Liu H, Tian HG and Li YF. Comparison of new hybrid FEEMD-MLP, FEEMD-ANFIS, wavelet packet-MLP and wavelet packet-ANFIS for wind speed predictions. *Energy Convers Manage* 2015; 89: 1–11.
38. Liu D, Wang J and Wang H. Short-term wind speed forecasting based on spectral clustering and optimized echo state networks. *Renew Energy* 2015; 78: 599–608.
39. Cadenas E and Rivera W. Wind speed forecasting in three different regions of Mexico, using a hybrid ARIMA-ANN model. *Renew Ener* 2010; 35: 2732–2738.
40. Liu H, Chen C, Tian H, et al. A hybrid model for wind speed prediction using empirical mode decomposition and artificial neural networks. *Renew Ener* 2012; 48: 545–556.
41. Su Z, Wang J, Lu H, et al. A new hybrid model optimized by an intelligent optimization algorithm for wind speed forecasting. *Ener Convers Manage* 2014; 85: 443–452.
42. Fei S-W and He Y. Wind speed prediction using the hybrid model of wavelet decomposition and artificial bee colony algorithm-based relevance vector machine. *Electr Power Ener Syst* 2015; 73: 625–631.
43. Liu H, Tian H-Q, Pan D-F, et al. Forecasting models for wind speed using wavelet, wavelet packet, time series and artificial neural networks. *Appl Ener* 2013; 107: 191–208.
44. Liu D, Niu D, Wang H, et al. Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm. *Renew Ener* 2014; 62: 592–597.
45. Rahmani R, Yusof R and Ismail N. A new metaheuristic algorithm for global optimization over continuous search space. *ICIC Int* 2015; 9(5): 1335–1340.
46. Said GA, Mahmoud AM and El-Horbaty E-S. A comparative study of meta-heuristic algorithms for solving quadratic assignment problem. *Int J Adv Comp Sci*

- Appl Comput* 2014; 5(1). <https://arxiv.org/ftp/arxiv/papers/1407/1407.4863.pdf>
47. Mahrami M, Rahmani R, Seyedmahmoudian M, et al. A hybrid metaheuristic technique developed for hourly load forecasting. *Complexity* 2016; 21(Suppl. 1): 521–532.
 48. Rahmani R and Yusof R. A new simple, fast and efficient algorithm for global optimization over continuous search-space problems: radial movement optimization. *Appl Mathemat Computat* 2014; 248: 287–300.
 49. Seyedmahmoudian M, Horan B, Rahmani R, et al. Efficient photovoltaic system maximum power point tracking using a new technique. *Energies* 2016; 9: 147.
 50. Kennedy J and Eberhart RC. Particle swarm optimization. In: *IEEE international conference on neural networks*, Perth, WA, Australia, 27 November–1 December 1995, vol. 4, pp. 1942–1948. New York: IEEE.
 51. Kennedy J, Eberhart RC and Shi Y. *Swarm intelligence*. Burlington, MA: Morgan Kaufmann Publishers, 2001, p. 512.
 52. Parsopoulos KE and Vrahatis MN. *Particle swarm optimization and intelligence: advances and applications*. New York: Information Science Reference, 2010, p. 310.
 53. Eberhart RC and Kennedy J. new optimizer using particle swarm theory. In: *Proceedings of the sixth international symposium on micro machine and human science (MHS 95)*, Nagoya, Japan, 4–6 October 1995.
 54. Mirjalili S, Mirjalili SM and Lewis A. Let a biogeography-based optimizer train your multi-layer perceptron. *Informat Sci* 2014; 269: 188–209.
 55. Mirjalili S, Mirjalili SM and Yang XS. Binary bat algorithm. *Neural Comp Applicat* 2014; 25(3–4): 663–681.
 56. Mirjalili S. The ant lion optimizer. *Adv Eng Software* 2015; 83: 80–98.