Energy Conversion and Management 87 (2014) 10–18

Contents lists available at ScienceDirect

Energy Conversion and Management

journal homepage: www.elsevier.com/locate/enconman

Feature selection in wind speed prediction systems based on a hybrid coral reefs optimization – Extreme learning machine approach

S. Salcedo-Sanz [a,*], A. Pastor-Sánchez [a], L. Prieto [b], A. Blanco-Aguilera [c], R. García-Herrera [c,d]

[a] Department of Signal Processing and Communications, Universidad de Alcalá, Alcalá de Henares, Spain
[b] Department of Energy Resource, Iberdrola, Spain
[c] Department of Atmospheric Physics, Universidad Complutense de Madrid, Spain
[d] IGEO, Instituto de Geociencias (UCM-CSIC), Madrid, Spain

ARTICLE INFO

Article history:
Received 28 August 2013
Accepted 15 June 2014
Available online 18 July 2014

Keywords:
Short term wind speed prediction
Feature selection problem
Coral reefs optimization algorithm
Extreme learning machines

ABSTRACT

This paper presents a novel approach for short-term wind speed prediction based on a Coral Reefs Optimization algorithm (CRO) and an Extreme Learning Machine (ELM), using meteorological predictive variables from a physical model (the Weather Research and Forecast model, WRF). The approach is based on a Feature Selection Problem (FSP) carried out with the CRO, that must obtain a reduced number of predictive variables out of the total available from the WRF. This set of features will be the input of an ELM, that finally provides the wind speed prediction. The CRO is a novel bio-inspired approach, based on the simulation of reef formation and coral reproduction, able to obtain excellent results in optimization problems. On the other hand, the ELM is a new paradigm in neural networks' training, that provides a robust and extremely fast training of the network. Together, these algorithms are able to successfully solve this problem of feature selection in short-term wind speed prediction. Experiments in a real wind farm in the USA show the excellent performance of the CRO–ELM approach in this FSP wind speed prediction problem.

1. Introduction

Wind power is currently the most important renewable energy source in the world in terms of annual growing and economic impact [1,2]. The installed wind power worldwide by the end of 2013 reached a total of 318 GW, with a few leading countries betting for this technology: China (91 GW), the USA (61 GW), Germany (34 GW), Spain (23 GW) or India (20 GW) [3], and many others in which wind energy is considered as the future source of alternative energy out of conventional sources, such as Denmark (25% of wind energy penetration), Portugal (16%), Ireland (12%), Italy (4%) or France (3%). This wind energy booming around the world has brought new problems in the management and maintenance of wind farm facilities [4]. One of this important problems is the integration of wind energy in the energy transportation network, where the prediction of the generated power in wind farms is a key problem, influenced by the variability of the wind in the short and medium terms. Thus, wind speed prediction is a basic task performed in all wind farms facilities as part of their operation management.

There are two types of approaches that have been used to carry out wind speed prediction in wind farm facilities. First, pure statistical approaches consider only previous wind speed series in one or several towers to construct a predictor for the wind speed in the near future. These approaches sometimes include meteorological variables measured at the prediction area, in order to enhance the wind speed prediction. In any case, these approaches do not take into account the atmospheric dynamics in order to make the wind speed prediction. In the last few years, many different statistical approaches have been applied to wind speed prediction, including linear prediction models [5], classical Box–Jenkins methodologies such as auto-regressive models [6] and other time series analysis such as the Mycielski algorithm [7], different clustering algorithms [8,9], and several modern computational approaches such as neural networks [10–13], neural networks ensembles [14], Bayesian methods [15], support vector machines [16,17], or combinations of different statistical models: neural networks and auto-regressive models [18], auto-regressive models and Kalman filtering [19], neural networks and Markov models [20], and wavelets and neural approaches [21,22].

* Corresponding author. Address: Department of Signal Processing and Communications, Universidad de Alcalá, 28871 Alcalá de Henares, Madrid, Spain. Tel.: +34 91 885 6731; fax: +34 91 885 6699.
E-mail address: sancho.salcedo@uah.es (S. Salcedo-Sanz).

http://dx.doi.org/10.1016/j.enconman.2014.06.041
0196-8904/

One alternative to this pure statistical approach is to consider pure physical models, usually meso-scale models, to make the wind speed prediction [23,24]. The main issue with this approach is the reduced spacial accuracy of the meso-scale models, conditioned by the spacial domains defined in the models. Also, the necessity of including different parameterizations in the models makes difficult the application of these methods in a wind farm in operation.

There is a third alternative, consisting in the combination of physical and statistical models [25,26]. The physical models can be global, meso-scale or even local, taking into account the specific local orography of the wind farm [27,28]. On the other hand, statistical models are usually included in these prediction systems to process the output of the physical models, and it has been shown that they produce a significant improvement in the prediction when compared with purely physical (and of course purely statistical) approaches [29–31]. One characteristic of these hybrid physical–statistical models is that the physical models produce a large number of different meteorological variables, which can be used as inputs for the statistical methods. In fact, even if we consider a reduced-size grid for the physical models, the number of available meteorological variables (from a given physical model), is huge, and some variable selection is needed. In most cases, the variable selection is carried out by randomly choosing a few meteorological variables with some experimental criteria [26,29], but it can be seen that a correct study on the input variables of statistical methods would improve the quality of the results obtained. This process of feature reduction is known in artificial intelligence as a Feature Selection Problem (FSP). Recently, different works specific on FSP in wind speed prediction have been discussed in the literature. In [32,33] Particle Swarm Optimization (PSO) and differential evolution algorithms, together with a k-nearest neighbors approach, are proposed to select the best variables in a wind speed prediction problem. A classical neural network (multi-layer perceptron) is used as statistical approach to improve the outcome of a physical model. The authors show how the PSO approach obtains the best results in terms of prediction error in several wind farms in Germany. More recently, in [34] a genetic algorithm is applied to the selection of the best set of features to feed a neural network in a wind speed prediction problem. The authors show the goodness of their proposal in data from several wind farms in India, obtaining improvements over the prediction system without feature selection. Recently, in [35] a PSO is proposed to optimize the main parameters of a neural network, including the training set length, in a problem of short term wind speed prediction. Experiments in a zone of high wind speed resource in the north west of China showed the accuracy of this proposal.

Even though these hybrid physical–statistical approaches have shown good results in wind speed prediction problems, there are several methodological problems on them that must be solved. First, different optimization algorithms can obtain better results than those tested up until now. Second, there is a serious problem of computational cost associated to this problem, that must be treated in order to improve the results. This paper is focussed solving the two main drawbacks of feature selection in wind speed prediction mentioned before. We therefore deal with a problem of feature selection in a hybrid wind speed prediction system based on a physical model with an statistical final approach (a fast-training neural network). The novelties and contributions of this paper are the following: First, we propose a new hybrid physical–statistical algorithm for a problem of short term wind speed prediction. The physical model is the WRF [23], whereas the statistical approach is a novel Coral Reef Optimization (CRO) [36] with an Extreme Learning Machine (ELM) [37]. The CRO approach is recently proposed meta-heuristic, that has shown very good performance in other optimization problems. In this case we use the CRO approach to select the best set of meteorological variables from the WRF, in terms of the prediction error obtained with the ELM network. We have chosen ELM as final regressor because it is able to provide excellent results within a very short computation time. Second, note that we state the wind speed prediction problem in a wind farm as a FSP. Thus, we consider a novel formulation of the problem, where the performance of the statistical approach (ELM) depends on the set of variables selected out of the physical model (WRF in this case). This requires a novel encoding of the FSP in the CRO, that is also a contribution of this work. Finally, we show the performance of the proposal in a real problem of wind speed prediction in a wind farm located at the west coast of the USA, showing the good performance of the proposed approach by means of comparison with an alternative prediction system based on a classical evolutionary algorithm.

The rest of this article is structured as follows: next section briefly introduces the feature selection problem in a formal way, describing the different existing methodologies to solve this problem. Section 3 presents the CRO algorithm used in this paper to tackle the feature selection problem in wind speed prediction. In this section we also describe the ELM network used in this work. Section 5 presents the experimental part of the paper, where the performance of the proposed approach is evaluated. Finally, Section 6 closes the paper with some concluding remarks.

2. The feature selection problem

Feature selection is an important task in supervised classification and regression problems because irrelevant features, used as part of the training procedure, can increase the cost and running time of a prediction system, and make its generalization performance poorer [38].

In its more general form, the FSP for a learning problem from samples can be addressed in the following way: given a set of labeled data points $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)$, where $\mathbf{x}_i \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$, choose a subset of m features $(m < n)$, that achieves the lowest error in the prediction of y_i.

There are two different approaches to the Feature Selection Problem (FSP). The first method tries to identify an appropriate set of features, independently of its classification performance, which preserve most of the information provided by the original data. This approach is known as filter method for feature selection [38]. Fig. 1(a) shows an outline of the filter method for feature selection. The second approach directly selects a subset of m features out of the total available in such a way that the performance of the classifier is improved or, at least, is not degraded. This method, known as wrapper method, is more powerful than filter methods, but it is also computationally more demanding [39,40]. Fig. 1(b) shows an outline of the wrapper method for feature selection. The search of the best feature subset can be performed by means of any search algorithm like hill-climbing, greedy or genetic algorithms.

All the previously discussed approaches to feature selection in wind speed prediction problems [32–34] deal with some kind of wrapper methods, where the regressor is a neural network and the search algorithm depends on the approach (particle swarm, differential evolution or genetic algorithms). In this paper we also study a wrapper approach for feature selection, formed by a CRO as searching strategy and an ELM to predict the wind speed. Next sections describe the different algorithms of the system.

3. The coral reefs optimization algorithm

The CRO is a novel meta-heuristic search approach based on corals' reproduction and coral reefs formation, proposed in [36]. Basically, the CRO is based on the artificial modeling of a coral reef,

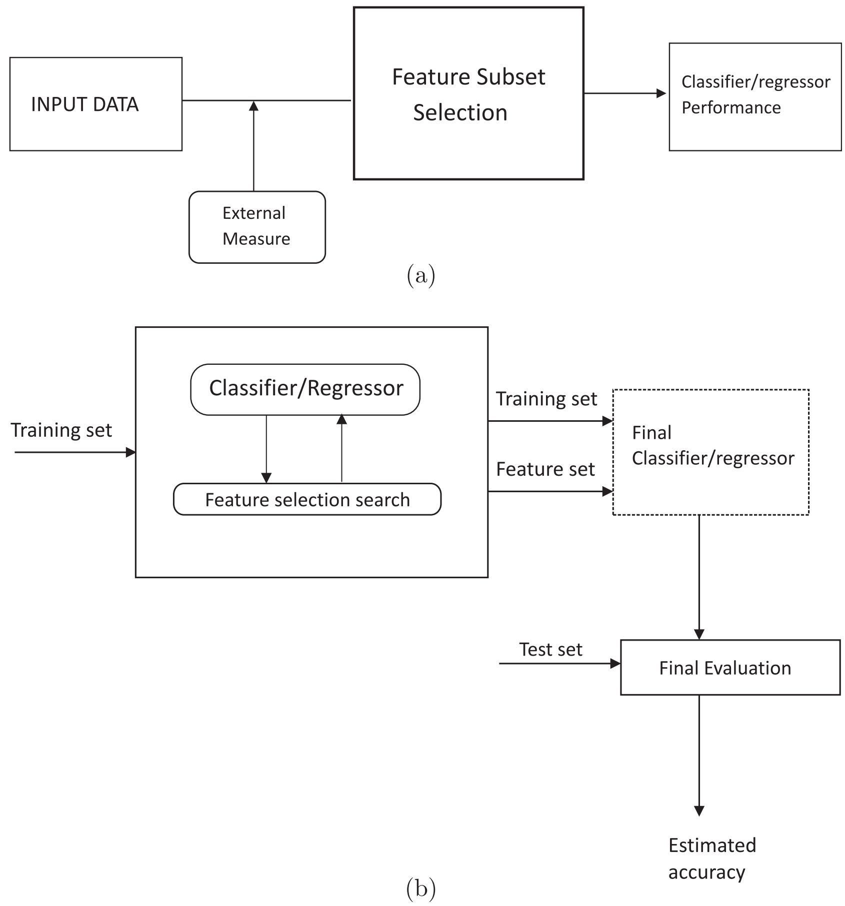

Fig. 1. Strategies in FSP; (a) filter methods and (b) wrapper methods.

Λ, consisting of a $\mathcal{N} \times \mathcal{M}$ square grid. We assume that each square (i,j) of Λ is able to allocate a coral (or colony of corals) $\Xi_{i,j}$, representing a solution to a given optimization problem, which is encoded as a string of numbers in a given alphabet $\mathcal{I}$. The CRO algorithm is first initialized at random by assigning some squares in Λ to be occupied by corals (i.e. solutions to the problem) and some other squares in the grid to be empty, i.e. holes in the reef where new corals can freely settle and grow in the future. The rate between free/occupied squares in Λ at the beginning of the algorithm is an important parameter of the CRO algorithm, which is denoted as ρ, and note that $0 < \rho < 1$. Each coral is labeled with an associated *health* function $f(\Xi_{ij}) : \mathcal{I} \rightarrow \mathbb{R}$, that represents the problem's objective function. The CRO is based on the fact that reef will progress, as long as healthier (stronger) corals (which represent better solutions to the problem at hand) survive, while less healthy corals perish.

After the reef initialization described above, a second phase of reef formation is artificially simulated in the CRO algorithm: a simulation of the corals' reproduction in the reef is done by sequentially applying different operators. This sequential set of operators is then applied until a given stop criteria is met. Several operators to imitate corals' reproduction are defined, among them: a modeling of corals' sexual reproduction (broadcast spawning and brooding), a model of asexual reproduction (budding), and also some catastrophic events in the reef, i.e. polyps depredation. After the sexual and asexual reproduction, the set of larvae formed (new solutions to the problem), try to locate a place to grow in the reef. It could be in a free space, or in an occupied once, by fighting against the coral actually located in that place. If larvae are not successful in locating a place to grow in a given number of attempts, they are depredated in this phase. Fig. 2 illustrates the flow diagram of the CRO algorithm referencing the two CRO phases (reef initialization and reef formation), along with all the operators described above.

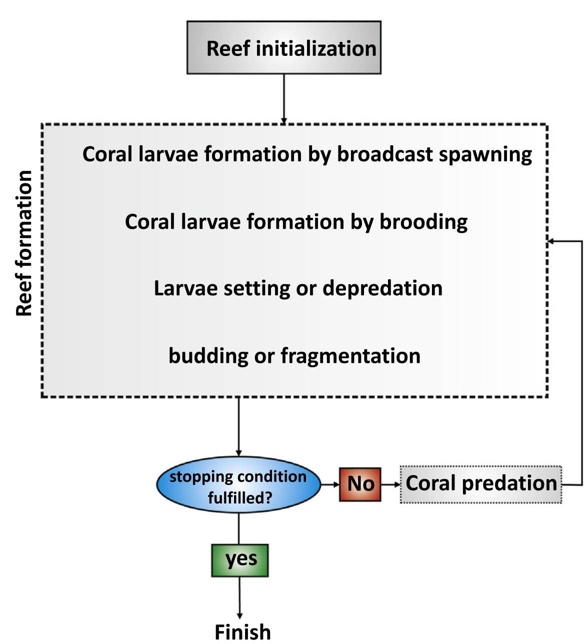

Fig. 2. Flow diagram of the proposed CRO algorithm.

1. *Broadcast Spawning (external sexual reproduction)*: the modeling of coral reproduction by *broadcast spawning* consists of the following steps:
   1.a. In a given step k of the reef formation phase a fraction of the existing corals is selected uniformly at random to be broadcast spawners. This fraction will be denoted as F_b. Corals that are not selected to be broadcast spawners (i.e. $1 - F_b$) will reproduce by brooding later on in the algorithm.
   1.b. Couples are selected from a pool of broadcast spawner corals in step k. Each of such couples will form a coral larva by sexual crossover, which is then released out to the water. Note that, once two corals have been selected to be the parents of a larva, they are not chosen anymore in step k (i.e. two corals are parents only once in a given step). These couple selection can be done uniformly at random or by resorting to any fitness proportionate selection approach (e.g. roulette wheel).
2. *Brooding (internal sexual reproduction)*: as previously mentioned, at each step k of the reef formation phase in the CRO algorithm, the fraction of corals that will reproduce by brooding is $1 - F_b$. The brooding modeling consists of the formation of a coral larva by means of a random mutation of the brooding-reproductive coral (self-fertilization considering hermaphrodite corals). The produced larva is then released out to the water in a similar fashion than that of the larvae generated in step 1.b.
3. *Larvae setting*: once all the larvae are formed at step k either through broadcast spawning (1) or by brooding (2), they will try to set and grow in the reef. First, the health function of each coral larva is computed. Second, each larva will randomly try to set in a square (i,j) of the reef. If the square is empty (free space in the reef), the coral grows therein no matter the value of its health function. By contrast, if a coral is already occupying the square at hand, the new larva will set only if its health function is better than that of the existing coral. We define a number κ of attempts for a larva to set in the reef: after κ unsuccessful tries, it will be depredated by animals in the reef.
4. *Asexual reproduction*: in the modeling of asexual reproduction (budding or fragmentation), the overall set of existing corals in the reef are sorted as a function of their level of healthiness (given by $f(\Xi_{ij})$), from which a fraction F_a duplicates itself and tries to settle in a different part of the reef by following the setting process described in Step 3. Note that a maximum number of identical corals (μ) are allowed in the reef.
5. *Depredation in polyp phase*: corals may die during the reef formation phase of the CRO algorithm. At the end of each reproduction step k, a small number of corals in the reef can be depredated, thus liberating space in the reef for next coral generation. The depredation operator is applied with a very small probability P_d at each step k, and exclusively to a fraction F_d of the worse health corals in Λ.

4. Feature selection with the CRO algorithm

The FSP tackled in this paper has the following description. First, we consider a grid Ω formed by $N \times N$ nodes and a given measuring tower $\mathfrak{M}$. We consider a time series of wind speed values in $\mathfrak{M}$, and a times series of M meteorological variables (features) in each node of the grid, obtained from a given physics-based prediction model. The wind speed series in $\mathfrak{M}$, and the meteorological series in the points of the grid are synchronized in time. Note that for large values of M, the number of available meteorological variables is huge (may be over 5000). The problem consists of predicting the wind speed in $\mathfrak{M}$ by using the predictive meteorological variables of the grid points. Fig. 3 shows an example of a grid Ω and measuring tower $\mathfrak{M}$. We consider then a fix number m of final meteorological variables (out of the total $n = M \times (N \times N)$) to do the wind speed prediction. In this case we have carried out experiments with different number of fixed variables m, so the objective of the problem is to obtain the best set of m variables that provides the best performance of the system in terms of wind speed prediction.

With this in mind, the encoding of each coral Ξ (problem's solution) in the CRO is the following: each meteorological variable included in the prediction system needs a total of four parameters to be identified, (x, y, id, ma), where x stands for the x-coordinate in the grid, y stands for the y-coordinate in the grid, id stands for the variable identifier and ma is a binary variable in such a way that a 1 means that we consider a moving average of the series of that variable, and a 0 means that such a moving average is not considered. The final encoding of a coral in the algorithm is therefore a $(m \times 4)$-length vector:

$$\Xi = [x_1, y_1, id_1, ma_1, \ldots, x_m, y_m, id_m, ma_m]. \tag{1}$$

4.1. Objective function: extreme learning machines

The ELM is a novel and fast learning method based on the structure of multi-layer perceptrons, recently proposed in [37] and applied thereafter to a large number of classification and regression problems [41–43]. The ELM's structure is similar to the network given in Fig. 4. The most significant characteristic of the ELM training is that it is carried out just by randomly setting the network weights, and then obtaining the inverse of the hidden-layer output matrix. The advantages of this technique are its simplicity, which makes the training algorithm extremely fast, and also its outstanding performance when compared to avantgarde learning methods, usually better than other established approaches such as classical multi-layer perceptrons or support vector machines. Moreover, the universal approximation capability of the ELM network, as well as its classification capability, have been already proven [44,45].

The ELM training method can be defined in the following way: Given a training set $\aleph \triangleq \{(\mathbf{x}_i, \mathbf{t}_i) | \mathbf{x}_i \in \mathbb{R}^n, \mathbf{t}_i \in \mathbb{R}, i = 1, \ldots, N_T\}$, an activation function $g(x)$ and a number of hidden nodes $(\widetilde{N})$, the ELM algorithm is summarized in a number of steps:

1. Randomly assign inputs weights $\mathbf{w}_i$ and bias b_i, with $i = 1, \ldots, \widetilde{N}$.
2. Calculate the $N_T \times \widetilde{N}$ hidden-layer output matrix $\mathbf{H}$, defined as

$$\mathbf{H} \triangleq \begin{bmatrix} g(\mathbf{w}_1\mathbf{x_1} + b_1) & \cdots & g(\mathbf{w}_{\widetilde{N}}\mathbf{x_1} + b_{\widetilde{N}}) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_1\mathbf{x}_{\mathbf{N_T}} + b_1) & \cdots & g(\mathbf{w}_{\widetilde{N}}\mathbf{x}_{\mathbf{N_T}} + b_{\widetilde{N}}) \end{bmatrix}. \tag{2}$$

3. Calculate the output weight vector β as

$$\beta = \mathbf{H}^{\dagger}\mathbf{T}, \tag{3}$$

   where $\mathbf{H}^{\dagger}$ stands for the Moore–Pennrose inverse of matrix $\mathbf{H}$ [37], and $\mathbf{T} \triangleq [t_1, \ldots, t_{N_T}]^T$ is the training output vector.

Note that the number of hidden nodes $\widetilde{N}$ is a free parameter of the ELM training, and must be estimated to obtain good results. Usually, scanning a range of $\widetilde{N}$ values is the most practical solution for this problem. It is well known that the ELM is an algorithm with a low computational complexity because it just involves the calculation of the output weights by means of the Moore–Pennrose matrix and other minor calculations. Because of their excellent performance along with their extreme fast training time, ELMs are perfect for hybrid algorithms requiring fast classifiers or regressors, such as in the case of feature selection. In this case, the ELM algorithm is used to calculate the Mean Square Error (MSE) of the wind speed prediction. MSE value will be used as coral health (objective) function in the CRO algorithm.

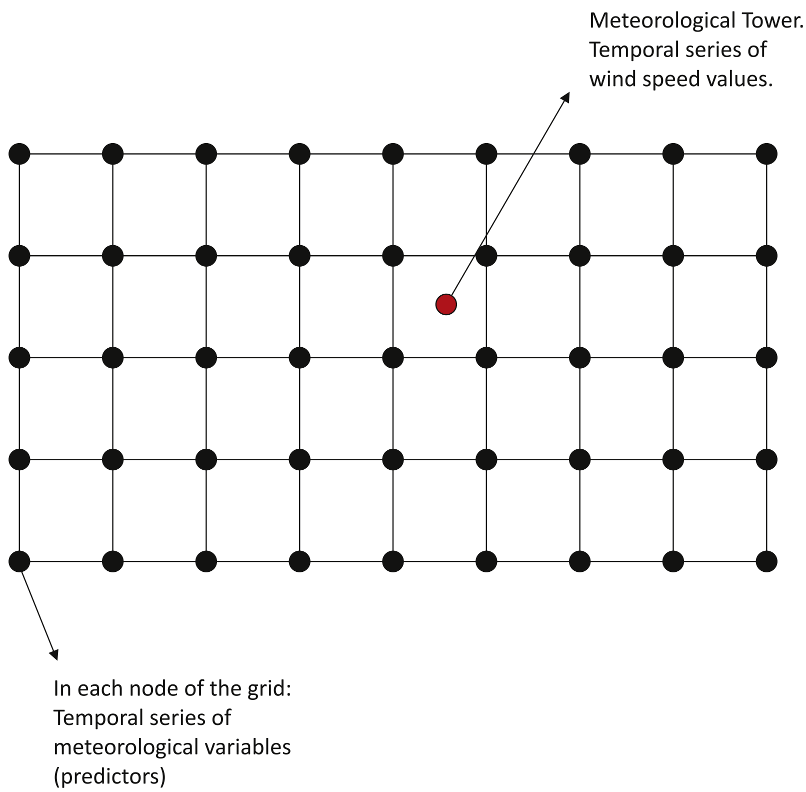

Fig. 3. Example of the grid and measuring tower in a wind farm.

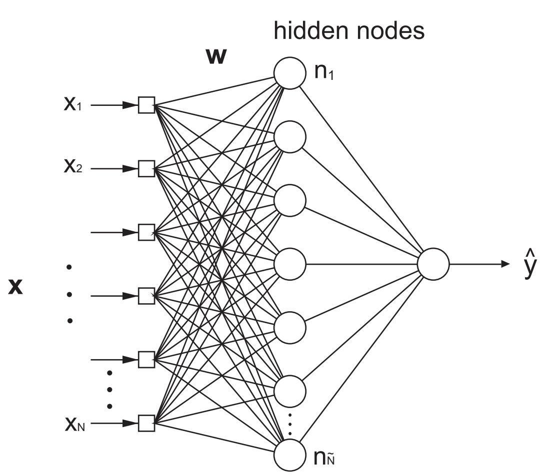

Fig. 4. Outline of the extreme learning machine structure.

5. Experimental part

In order to test the performance of the proposed CRO–ELM algorithm for short-term wind speed prediction, we have carried out a number of experiments with real wind speed data from a measuring tower ($\mathfrak{M}$) in a wind farm in USA (see Fig. 5). In the following sections we describe in detail the data used to evaluate the CRO–ELM performance, the predictive variables considered in this case, as well as a brief description of alternative algorithms we have used to contextualize the CRO–ELM analysis.

5.1. Data used, variables considered and methodology

One year of hourly 10 m wind speed data (01/03/2007–29/02/2008) is considered. An 11×11 grid surrounding the measuring tower is taken into account, and in each node of the grid, a series of 27 meteorological variables (at different height levels) is considered. Table 1 shows the predictive meteorological variables considered. Variables in the time considered for analysis (01/03/2007–29/02/2008) have been obtained with a meso-scale WRF model in backcast or hindcast mode, using the National Center for Environmental Prediction and National Center of Atmospheric Research (NCEP/NCAR) global Reanalysis dataset. Note that there are 27 predictive variables, some of them are direct outputs of the meso-scale model and some others correspond to derived variables. As mentioned, we also consider the possibility of a moving average on the variable series, so we finally have $M = 54$ possible variables to be selected in each grid point. Thus, the total number of variables involved in the problem is $n = M \times (N \times N)) = 54 \cdot 121 = 6534$ variables. We have first split the available data into a training (75% of the data) and a test set (25%). After this first splitting of the data, the following methodology has been carried out in order to obtain a significant solution out of the CRO–ELM algorithm: as has been mentioned before, MSE value will be used as coral health (objective) function in the CRO algorithm. However, this MSE must be calculated in the CRO considering only the training data. In order to obtain an objective measure of MSE which provides then the best generalization of the algorithm, we have included a procedure of n cross-validation, in which the training data is split again into n sets, and the ELM is trained with $n - 1$ of these sets and tested in the remaining one, in such a way that all the sets are used as test set. The MSE provided as health function of the coral will be the average of all the MSE obtained for each of the sets serving as test set. In a final step, after the best coral has been obtained in the CRO, we can obtain its final associated MSE in the test set (25% of the original data, training the ELM with the 75% of these original data), as the final result of the CRO–ELM. Note that all the results shown in this paper are referred to this final MSE in the test set. Finally, just to highlight once more that the problem we face is therefore to obtain

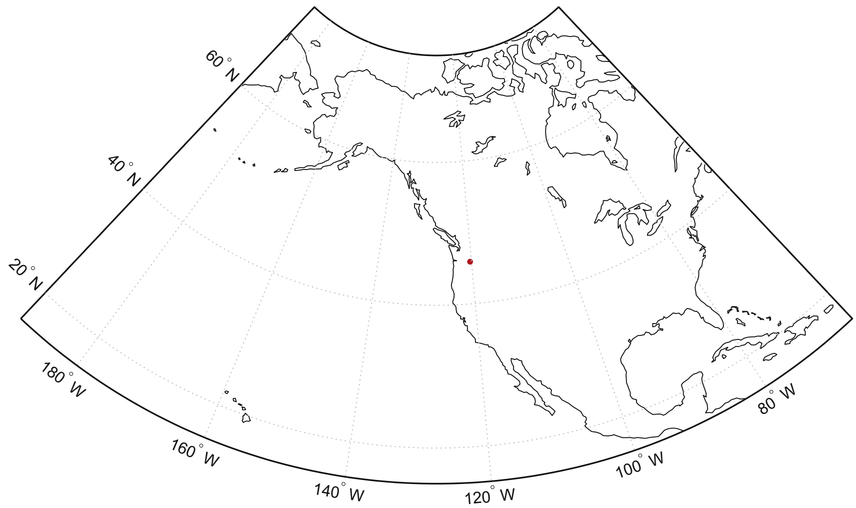

Fig. 5. Situation of the wind farm considered in this work.

Table 1
Predictive meteorological variables used in the short-term wind speed prediction problem considered.

id #	Meteorological variable
Direct measures	
0–5	Wind speed and direction at different heights (0 m, 10 m, 20 m, 50 m, 80 m, 100 m)
6–10	Wind direction at different heights (0 m, 10 m, 20 m, 50 m, 80 m)
11–13	Temperature at different heights (0 m, 2 m, 20 m)
14	Specific humidity (2 m)
15	Sea level pressure
16	Long wave down radiation (0 m)
17	Short wave down radiation (0 m)
18	Precipitation
19–26	Function combinations of variables 0–18

the best set of m variables out of the 6534 possible that optimize the wind speed prediction done by the ELM algorithm. Note also that the value of m should be small in order to favor the ELM training.

5.2. Algorithms for comparison

We can establish two levels of comparison with the proposed CRO–ELM. First, we can evaluate the goodness of the CRO as global searcher. In order to do it, we have used an Evolutionary Algorithm (EA) [46] to solve the same FSP associated with short-term wind speed prediction. In this case, the structure of the algorithm for comparison is exactly the same that the proposed approach, substituting the CRO by an EA. A standard EA with two-points crossover and Gaussian mutation has been used in this comparison. The second level of comparison is the evaluation of the ELM as a regressor. In this case, note that the computation time of any candidate to substitute the ELM in the proposed approach must be extremely fast. Otherwise, the computation time of the algorithm could be completely unacceptable. Considering this important constraint, we have tested a linear regressor to substitute the ELM in this second comparison carried out, to form a CRO–LR algorithm. The linear regression is a well-known regression approach, which uses a linear model to estimate the dependent variable, in the following way:

$$y = \mathbf{X} \cdot \beta + \epsilon, \tag{4}$$

where **X** stands for the matrix of predictive variables, β is a vector of weights and **beta** the bias vector to complete the linear regression. Using this alternative, we can evaluate the goodness of the ELM as regressor in this problem of wind speed prediction.

5.3. Results

Table 2 shows the results obtained (in terms of MSE) by the CRO–ELM and EA–ELM approaches. The CRO–ELM algorithm obtains better results than the EA–ELM in all the experiments carried out, with a small error in wind speed prediction. The average improvement obtained with the CRO–ELM over the EA–ELM algorithm in terms of mean square error is about 2%. This may seem a small improvement, but it means an important improvement in terms of energy production and obtained revenue for an average wind farm, as we will show in the final discussion of this section. Note that the proposed wind speed prediction system is able to obtain accurate wind speed prediction, with a mean square error in the test set around 2.5 m/s. It is interesting that the best result obtained with the CRO–ELM contains $m = 9$ variables. This means that we have reduced the total 6534 initial possible variables to just 9, keeping the accuracy in the prediction. This implies an improvement in the wind speed prediction system in terms of computational complexity, since once the system is trained, we will have to consider 9 variables to obtain a good wind speed prediction. Fig. 6 illustrates the variables chosen by the CRO–ELM approach, and their location in the grid. As can be inferred from this plot, characteristics from the leftmost part of the grid seems to be dominant in the feature selection process carried out by the CRO algorithm. Interesting is also to remark that moving-average based variables (tagged with identifiers from 27 to 54) occur to be more frequent in the best solutions provided by the CRO than direct meteorological variables (tagged from 0 to 26).

In the second comparison carried out to evaluate the performance of the proposed algorithm, we analyze a comparison between the CRO–ELM and CRO–LR approaches. In this case we focus on the case of 9 final variables which is the one which obtained a better wind speed prediction. The result obtained by the CRO–LR is in this case 2.989 m/s in MSE. This results is poorer than the one obtained with the ELM, which seems to be more accurate with a similar computational cost. Fig. 7 shows the best CRO evolution obtained, considering cross-validation MSE with the ELM and LR regressors. We can also have a visual shot of the CRO–ELM performance in this problem by comparing the best wind speed prediction obtained against the real wind speed. Fig. 8 shows such a comparison. Note that the proposed CRO–

Table 2
Results obtained with the CRO–ELM and EA–ELM in the FSP problem associated with short-term wind speed prediction.

# Predictive variables	MSE CRO–ELM (m/s)	MSE EA–ELM (m/s)
4	2.573	2.612
5	2.565	2.594
6	2.552	2.589
7	2.530	2.582
8	2.543	2.567
9	2.503	2.556
10	2.505	2.567
11	2.509	2.577
12	2.517	2.579
13	2.515	2.587
14	2.511	2.589
15	2.534	2.600

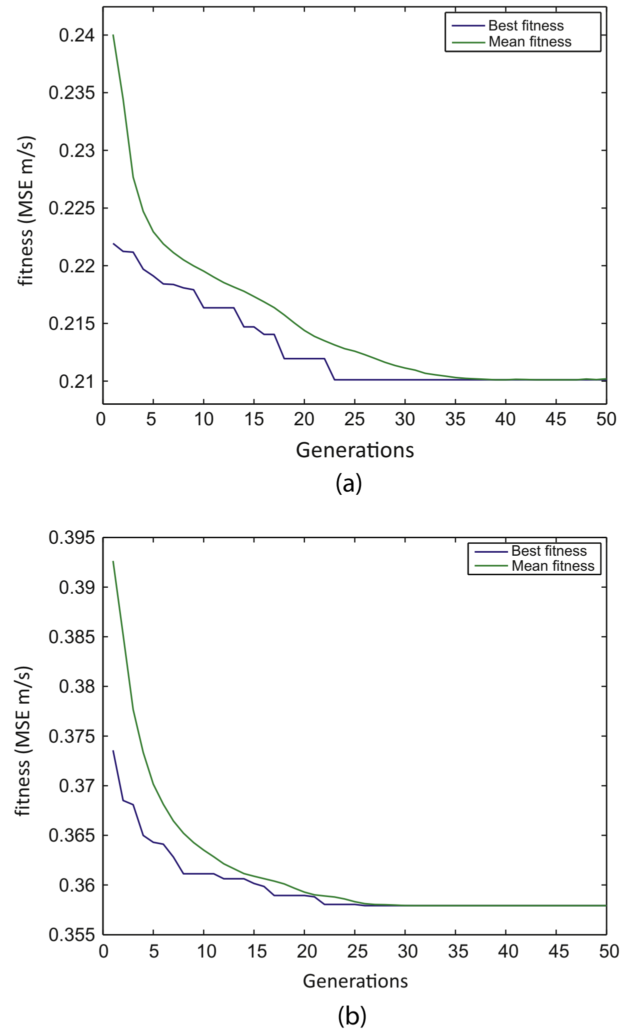

Fig. 7. CRO–ELM and CRO–LR evolution; (a) CRO–ELM and (b) CRO–LR.

ELM obtains an accurate reconstruction of the wind speed, missing some ramps and maximum/minimum wind speed values, but following quite well the wind speed trend in general.

5.4. Further analysis and discussion

In the previous section we have analyzed the performance of the ELM and LR evolution (using a CRO and EA algorithms), in order to fix a small number of predictive variables in a problem of wind speed forecast. As has been mentioned, the election of the ELM or LR algorithms is useful because the wrapper feature selection requires low computational approaches to be hybridized with the global search algorithms. There are, however, powerful algorithms for regression that could produce excellent results in wind speed prediction. One of this approaches is the Support Vector Regression (SVMr) algorithm [47], that is known to be one of the most accurate existing regressor algorithms, and it has been successfully applied to wind speed prediction previously [31,30]. On the other hand, this algorithm involves a high computation time, even harder if SVMr parameters are sought previously to its application to the problem (in this case we obtain the SVMr parameters by using a grid search approach). Thus, it is not directly applicable in hybridization with the CRO, but, we could, of course, test its performance by applying it to the features selected by the CRO–ELM or CRO–LR algorithms. The idea is to select the best reduced set of predictive feature with the CRO–ELM or CRO–LR, and then, using the SMVr to test this reduced set of features (training with the complete 75% of samples and testing the results in the remainder 25%). We have carried out this experiment, in the case of a set of 9 features. A process of SVMr parameters tuning following a grid search with bounds [49] has been carried out in a reduced set before launching the algorithm in the test set. Table 3 shows these results. Note that the SVMr approach applied to the features selected either by the CRO–ELM or CRO–LR provides better results than the ELM, in spite of the FSP has been driven with the latter.

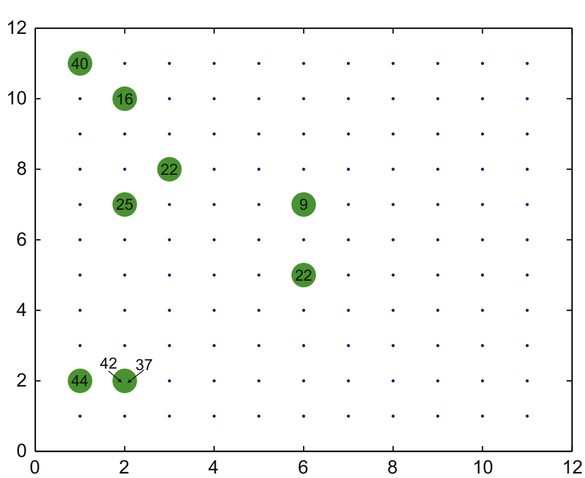

Fig. 6. Variables selected by the CRO–ELM algorithm in this problem of short-time wind speed prediction.

In order to analyze the impact that the improvement in wind speed prediction has in terms of economical revenues, we have carried out the following procedure: first, we have obtained the energy production series from the wind speed series, by using the model in [48]. This model considers the behavior of complete wind farms and not only isolated wind turbines. We have assumed an average wind farm with 50 MW, with standard losses and wind turbines availabilities. Due to the non-linearity of energy production with the wind, in this case, the MSE in energy is similar to the one obtained for the wind (about 2%). Considering the Spanish case, in which penalties for errors in wind energy production are different depending on the hour and there are differences in penalties for under and over estimation of production, an improvement of 2% in the wind speed prediction implies a revenue in the final wind energy price about 0.1% approximately. In Spain, for a standard 50 MW wind farm and a medium capacity of 26.5% this implies an extra income of 6500 Euros/year.

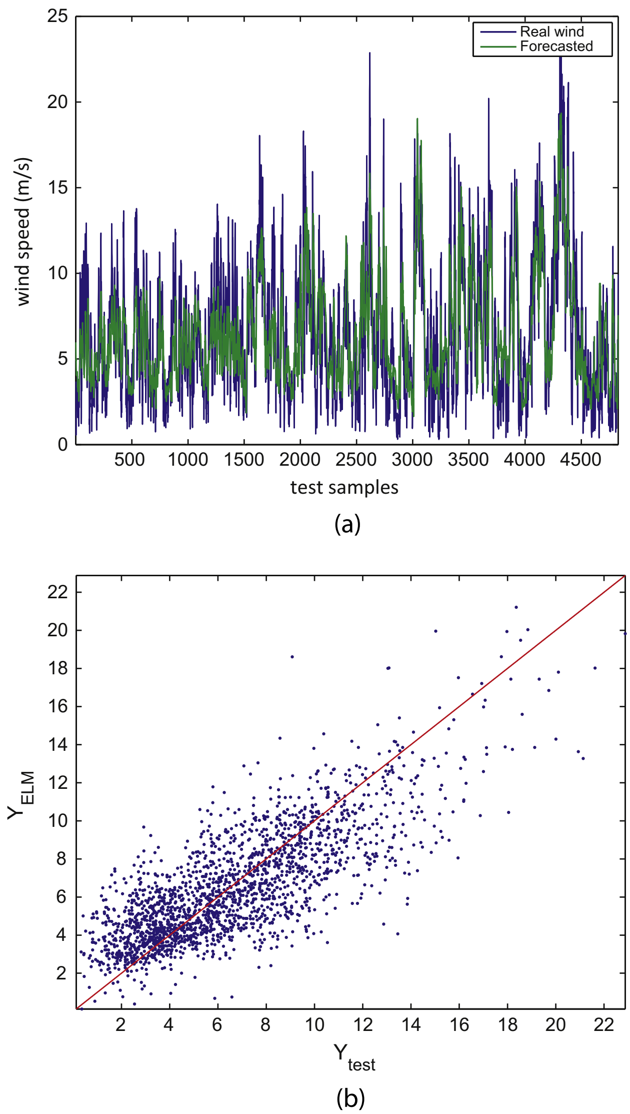

Fig. 8. Wind speed prediction obtained with the CRO–ELM algorithm and observed wind speed; (a) prediction in time and (b) dispersion graph.

Table 3
Results obtained with the ELM and SVMr approaches as prediction algorithms, using the features selected by the CRO–ELM and CRO–LR algorithms.

	ELM	SVMr
CRO–ELM	2.503	2.241
CRO–LR	2.679	2.436

6. Conclusions

This paper proposes a new hybrid physical–statistical approach for short-term wind speed prediction in wind farms. The proposed algorithm is formed by a new bio-inspired approach (the Coral Reefs Optimization algorithm, CRO), hybridized with a fast-training neural network (Extreme Learning Machine, ELM). One of the contributions of the problem is that the wind speed prediction problem has been stated in this work as a feature selection problem from the output of a meso-scale model, in such a way that the objective of the system is to obtain the best possible subset of features to be the input of a regressor (the ELM in this case), in terms of an error of prediction.

The performance of the system has been tested in a real problem of wind speed prediction in a wind farm sited on the west northern coast of the USA. The CRO–ELM approach has been tested there against the performance of a hybrid Evolutionary Algorithm with the ELM (EA–ELM) approach, in order to study what is the contribution of the CRO to the complete algorithm, and also with another hybrid approach formed by the CRO and a linear regressor approach (CRO–LR), in order to evaluate the action of the ELM in the algorithm. The results have shown that the CRO–ELM is the best approach over all tested in this work, obtaining the less error in wind speed prediction. Specifically, the average improvement obtained with the CRO–ELM over the EA–ELM algorithm in terms of mean square error is about 2%. The number of features that provided the best results in terms of error between the observed and the predicted wind speed was 9. Experiments with different number of features showed a worse performance of all the tested algorithms. A final analysis by using the features selected with the CRO–ELM into a different regressor (Support Vector Regressor algorithm in this case), has shown a better result than the ELM, even though the step of feature selection was carried out with the ELM. This indicates that feature selection process performed with the CRO is robust, and the features selected are significant in this paper of wind speed prediction.

Acknowledgements

This work has been partially supported by Iberdrola Renovables Energía S.A. This work has been also supported by Spanish Ministry of Economy, under project number ECO2010-22065-C03-02.

References

[1] Kaldellis JK, Zafirakis D. The wind energy (r)evolution: a short review of a long history. Renew Energy 2011;36(7):1887–901.
[2] Saidur R, Islam MR, Rahim NA, Solangi KH. A review on global wind energy policy. Renew Sustain Energy Rev 2010;14(7):1744–62.
[3] GWEC Global Wind Statistics 2013, Global Wind Energy Commission. <http://www.gwec.net/global-figures/graphs/>.
[4] Landberg L, Giebel G, Nielsen HA, Nielsen TS, Madsen H. Short-term Prediction – an overview. Wind Energy 2003;6(3):273–80.
[5] Riahy GH, Abedi M. Short term wind speed forecasting for wind turbine applications using linear prediction. Renew Energy 2008;33(1):35–41.
[6] Torres JL, García A, De Blas M, De Francisco A. Forecast of hourly average wind speed with ARMA models in Navarre (Spain). Sol Energy 2005;79:65–77.
[7] Hocaoglu FO, Fidan M, Gerek ON. Mycielski approach for wind speed prediction. Energy Convers Manage 2009;50(6):1436–43.
[8] Kusiak A, Li W. Short-term prediction of wind power with a clustering approach. Renew Energy 2010;35(10):2362–9.
[9] Yesilbudak M, Sagiroglu S, Colak I. A new approach to very short term wind speed prediction using *k*-nearest neighbor classification. Energy Convers Manage 2013;69:77–86.
[10] Barbounis TG, Theocharis JB. Locally recurrent neural networks for wind speed prediction using spatial correlation. Inform Sci 2007;177:5775–97.
[11] Li S, Wunch CD, O'Hair A, Giesselmann GM. Using neural networks to estimate wind turbine power generation. IEEE Trans Energy Convers 2001;16(3):276–81.
[12] Bilgili M, Sahin B, Yasar A. Application of artificial neural networks for the wind speed prediction of target stations using reference stations data. Renew Energy 2007;32:2350–60.
[13] Li G, Shi J. On comparing three artificial neural networks for wind speed forecasting. Appl Energy 2010;87(7):2313–20.
[14] Kusiak A, Li W. Estimation of wind speed: a data-driven approach. J Wind Eng Ind Aerodyn 2010;98(10–11):559–67.
[15] Jiang Y, Song Z, Kusiak A. Very short-term wind speed forecasting with Bayesian structural break model. Renew Energy 2013;50:637–47.
[16] Mohandes MA, Halawani TO, Rehman S, Hussain AA. Support vector machines for wind speed prediction. Renew Energy 2004;29:939–47.
[17] Zhou J, Shi J, Li G. Fine tuning support vector machines for short-term wind speed forecasting. Energy Convers Manage 2011;52:1990–8.
[18] Khashei M, Bijari M, Raissi-Ardali G. Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs). Neurocomputing 2009;72(4–6):956–67.
[19] Liu H, Tian H, Li YF. Comparison of two new ARIMA-ANN and ARIMA-Kalman hybrid methods for wind speed prediction. Appl Energy 2012;98:415–24.

[20] Pourmousavi Kani SA, Ardehali MM. Very short-term wind speed prediction: a new artificial neural network-Markov chain model. Energy Convers Manage 2011;52(1):738–45.
[21] Zhang W, Wang J, Wang J, Zhao Z, Tian M. Short-term wind speed forecasting based on a hybrid model. Appl Soft Comput 2013;13(7):3225–33.
[22] Liu H, Tian H, Pan D, Li Y. Forecasting models for wind speed using wavelet, wavelet packet, time series and Artificial Neural Networks. Appl Energy 2013;107:191–208.
[23] González-Mingueza C, Muñoz-Gutiérrez F. Wind prediction using weather research forecasting model (WRF): a case study in Peru. Energy Convers Manage 2014;81:363–73.
[24] Carvalho D, Rocha A, Gómez-Gesteira M, Silva-Santos C. WRF wind simulation and wind energy production estimates forced by different reanalyses: Comparison with observed data for Portugal. Appl Energy 2014;117:116–26.
[25] Costa A, Crespo A, Navarro J, Lizcano G, Madsend H, Feitosa E. A review on the young history of the wind power short-term prediction. Renew Sustain Energy Rev 2008;12(6):1725–44.
[26] Salcedo-Sanz S, Pérez-Bellido AM, Ortiz-Garcia EG, Portilla-Figueras A, Prieto L, Paredes D. Hybridizing the fifth generation mesoscale model with artificial neural networks for short-term wind speed prediction. Renew Energy 2009;34(6):1451–7.
[27] Landberg L. Short-term prediction of the power production from wind farms. J Wind Eng Ind Aerodyn 1999;80:207–20.
[28] Landberg L. Short-term prediction of local wind conditions. J Wind Eng Ind Aerodyn 2001;89:235–45.
[29] Salcedo-Sanz S, Pérez-Bellido AM, Ortiz-García EG, Portilla-Figueras A, Prieto L, Correoso F. Accurate short-term wind speed prediction by exploiting diversity in input data using banks of artificial neural networks. Neurocomputing 2009;72(4–6):1336–41.
[30] Salcedo-Sanz S, Ortiz-García EG, Pérez-Bellido AM, Portilla-Figueras A, Prieto L. Short term wind speed prediction based on evolutionary support vector regression algorithms. Expert Syst Appl 2011;38(4):4052–7.
[31] Ortiz-García EG, Salcedo-Sanz S, Pérez-Bellido AM, Gascón-Moreno J, Portilla-Figueras JA, Prieto L. Short-term wind speed prediction in wind farms based on banks of support vector machines. Wind Energy 2011;14(2):193–207.
[32] Jursa R. Variable selection for wind power prediction using particle swarm optimization. In: Proc of the 9th genetic and evolutionary computation conference (GECCO), London, England; 2007. p. 2059–65.
[33] Jursa R, Rohrig K. Short-term wind power forecasting using evolutionary algorithms for the automated specification of artificial intelligence models. Int J Forecast 2008;24(4):694–709.
[34] Gupta RA, Kumar R, Bansal AK. Selection of input variables for the prediction of wind speed in wind farms based on genetic algorithms. Wind Eng 2011;35(6): 649–60.
[35] Ren C, An N, Wang J, Li L, Hu B, Shang D. Optimal parameters selection for BP neural network based on particle swarm optimization: a case study of wind speed forecasting. Knowl-based Syst 2014;56:226–39.
[36] Salcedo-Sanz S, Del Ser J, Gil-López S, Landa-Torres I, Portilla-Figueras JA. The coral reefs optimization algorithm: a new metaheuristic algorithm for hard optimization problems. In: Proc of the 15th international conference on applied stochastic models and data analysis, (ASMDA), Mataró, Barcelona; 2013.
[37] Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing 2006;70(1):489–501.
[38] Blum A, Langley P. Selection of relevant features and examples in machine learning. Artif Intell 1997;97:245–71.
[39] Weston H, Mukherjee S, Chapelle O, Pontil M, Poggio T, Vapnik V. Feature selection for SVMs. In: Advances in NIPS 12. MIT Press; 2000. p. 526–32.
[40] Kohavi R, John GH. Wrappers for features subset selection. Int J Digital Libr 1997;1:108–21.
[41] Huang GB, Wang DH, Lan Y. Extreme learning machines: a survey. Int J Mach Learn Cybernet 2011;2(2):107–22.
[42] Huang GB, Chen L. Convex incremental extreme learning machine. Neurocomputing 2007;70:3056–62.
[43] Huang GB, Chen L. Enhanced random search based incremental extreme learning machine. Neurocomputing 2008;71:3460–8.
[44] Huang GB, Chen L, Siew CK. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Networks 2006;17(4):879–92.
[45] Huang GB, Zhou H, Ding X, Zhang R. Extreme learning machine for regression and multi-class classification. IEEE Trans Syst Man Cybernet Part B 2012;42(2):513–29.
[46] Eiben AE, Smith JE. Introduction to evolutionary computing. Springer-Verlag; 2003.
[47] Smola AJ, Schölkopf B. A tutorial on support vector regression. Stat Comput 2004;14:199–222.
[48] Norgaard P, Holttinen H. A multi-turbine power curve approach. In: Proc of the nordic wind power conference, vol. 1–2; 2004. p. 1–5.
[49] Ortiz-García EG, Salcedo-Sanz S, Pérez-Bellido AM, Portilla-Figueras JA. Improving the training time of support vector regression algorithms through novel hyper-parameters search space reductions. Neurocomputing 2009;72: 3683–91.