Neurocomputing 275 (2018): 818-828

DOI: https://doi.org/10.1016/j.neucom.2017.09.025

Bayesian Optimization of a Hybrid System for Robust Ocean Wave Features Prediction

L. Cornejo-Bueno[a], E. C. Garrido-Merchán[b], D. Hernández-Lobato[b], S. Salcedo-Sanz[a,c]

[a] *Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Madrid, Spain.*
[b] *Computer Science Department, Universidad Autónoma de Madrid, 28049, Madrid, Spain.*
[c] *Corresponding author: Sancho Salcedo Sanz. email: sancho.salcedo@uah.es*

Abstract

In the last years, Bayesian optimization (BO) has emerged as a practical tool for high-quality parameter selection in prediction systems. BO methods are useful for optimizing black-box objective functions that either lack an analytical expression, or are very expensive to evaluate. In this paper we show that BO can be used to obtain the optimal parameters of a prediction system for problems related to ocean wave features prediction. Specifically, we propose the Bayesian optimization of a hybrid Grouping Genetic Algorithm for attribute selection combined with an Extreme Learning Machine (GGA-ELM) approach for prediction. The system uses data from neighbor stations (usually buoys) in order to predict the significant wave height and the wave energy flux at a goal marine structure facility. The proposed BO methodology has been tested in a real problem involving buoys data in the Western coast of the USA, improving the performance of the GGA-ELM without a BO approach.

Keywords: Ocean waves features; Prediction System; Bayesian optimization.

1. Introduction

The accurate prediction of waves features plays a key role in different ocean engineering–related activities, such as safe ship navigation [1, 2], the design of marine structures [3, 4], *e.g.*, oil platforms and harbours, and in marine energy management problems [5, 6], like the proper operation of wave energy converters [7], among others. Thus, the topic has a clear impact on human safety, economics and clean energy production. One of the most important features to define the severity of a given ocean wave field is the significant wave height, H_{m_0}. H_{m_0} is usually estimated using in-situ sensors, such as buoys, recording time series of wave elevation information. Buoys provide reliable sea state information that characterizes the wave field in a fixed position (i.e. the mooring point). In addition, as buoys are anchored in a hostile media (the ocean), the probability that measuring problems (and therefore missing data) occur in situations of severe weather is very high [8]. Besides this, marine energy [9, 10] is currently one of the most promising sources of renewable energy, still minor at a global level, but playing a major role in several offshore islands [11, 12]. In this case, the accurate estimation of the wave energy flux P is relevant to characterize the wave energy production from Wave Energy Converters (WECs) facilities [13].

The research work on wave features prediction systems has been intense in the last years, with special incidence in machine learning approaches. One of the first works on this topic was the direct prediction of H_{m_0} using artificial neural networks in [14]. Improvements on this prediction system were further presented in [15]. Neural networks have also been applied to other problems of H_{m_0} and P prediction, such as [16], where H_{m_0} and P are inferred from observed wave records using time series neural networks. In [17] neural networks were applied to estimate the final breaking wave-height for laboratory-scaled and full-scaled ocean waves, showing that neural models are able to improve previously proposed empirical models for breaking waves-height estimation in terms of accuracy. In [18] a neural network was applied to estimate the wave energy resource in the northern coast of Spain. In [19] a hybrid genetic algorithm-adaptive network-based fuzzy inference system model was developed to forecast H_{m_0} and the peak spectral period at Lake Michigan. In [20] and [21] different hybrid algorithms mixed with an Extreme Learning Machine neural network were proposed for the estimation of H_{m_0} and P, in the context of marine energy applications. Alternative methods based on different computational approaches have been recently proposed. For example, in [22] different soft-computing techniques are tested for H_{m_0} prediction. Support Vector regression (SVR) has also been applied to marine energy related problems such as in [23]. Similarly, [24] and [25] proposed to feed SVR approaches with information from radar sources in order to obtain an accurate prediction of H_{m_0} and P features. Classification approaches have been applied in [26] to analyze and predict H_{m_0} and P ranges in buoys for marine energy applications. In [27], use of genetic programming for H_{m_0} reconstruction problems was proposed. Finally, in [28] fuzzy logic-based approaches were introduced for H_{m_0} prediction problems.

In this paper we test a BO methodology to improve the performance of a hybrid prediction system for wave features (H_{m_0} and P) prediction. Specifically, the prediction system was previously presented in [21], and it is formed by a Grouping Genetic Algorithm for feature selection, and an Extreme Learning Machine for carrying out the final energy flux prediction. This hybrid prediction system has a number of parameters that may affect its final performance, and need to be previously specified by the practitioner. Traditionally, these parameters have been manually tuned by a human expert, with experience in both the algorithm and the problem domain. However, it is possible to obtain better results by an automatic fine tuning of the prediction system's parameters. In this case, the parameters of GGA-ELM approach include the probability of mutation in the GGA or the number of neurons in the ELM hidden layer, among others. We propose then to use a Bayesian optimization (BO) approach to automatically optimize the parameters of the whole prediction system (GGA-ELM), with the aim of improving its performance in wave energy prediction problems. BO has been shown to obtain good results in the task of obtaining good parameter values for prediction systems [29]. In the paper we detail the basic prediction system considered and the BO methodology implemented, along with the improvements obtained in real problems of H_{m_0} and P prediction in the Western coast of the USA.

The rest of the paper is organized as follows: the next section details the calculation of the features of interest in ocean wave characterization, H_{m_0} and P in this case. Section 3 describes the main characteristics of the hybrid system to be optimized, which is formed by a GGA and an ELM for prediction. Section 4 presents the Bayesian optimization methodology applied in this case to optimize the prediction system considered. Section 5 presents the experimental part of the paper, where the Bayesian hybrid GGA-ELM approach is tested in a real problem of P prediction in the Western coast of the USA. Finally, Section 6 closes the paper exposing the conclusions of this work.

2. Wave features of interest: calculation of H_{m_0} and P

In the evaluation of marine systems it is essential to previously characterize as accurately as possible the wave features of the zone under study. For example, in a wave energy facility, it is necessary to characterize the amount of wave energy available at a particular location, which is given by features such as H_{m_0} and P. In order to obtain these features, it is necessary to focus on the water surface, and within the framework of the linear wave theory, the vertical wave elevation, $\eta(\mathbf{r}, t)$, at a point $\mathbf{r} = (x, y)$ on the sea surface at time t can be assumed as a superposition of different monochromatic wave components [30, 31]. This model is appropriate when the free wave components do not vary appreciably in space and time (that is, statistical temporal stationarity and spatial homogeneity can be assumed [31]).

In the model described, the concept of "sea state" refers to the sea area and the time interval in which the statistical and spectral characteristics of the wave do not change considerably (statistical temporal stationarity and spatial homogeneity). The features of a given sea state are then the combined contribution of all features from different sources. For example, the "wind sea" occurs when the waves are caused by the energy transferred between the local wind and the free surface of the sea. The "swell" is the situation in which the waves have been generated by winds blowing on another far area (for instance, by storms), and propagate towards the region of observation. Usually, sea states are the composition of these two pure states, forming multi-modal or mixed seas. In a given sea state, the wave elevation $\eta(\mathbf{r}, t)$ with respect to the mean ocean level can be assumed as a *zero-mean Gaussian stochastic process*, with statistical symmetry between wave maxima and minima. A buoy deployed at point $\mathbf{r}_B$ can take samples of this process, $\eta(\mathbf{r}_B, t_j)$ $j = 1, 2, \cdots, t_{\mathrm{MAX}}$, generating thus a time series of empirical vertical wave elevations. The Discrete Fourier Transform (DFT) of this sequence, using the Fast Fourier Transform (FFT) algorithm, allows for estimating the *spectral density* $S(f)$. Its spectral moments of order n can be computed as follows:

$$m_n = \int_0^\infty f^n S(f) df. \tag{1}$$

The Significant Wave Height (SWH) is defined as the average (in meters) of the highest one-third of all the wave heights during a 20-minute sampling period [32], and it has been widely studied. It can be calculated from the moment of order 0 in Equation (1), as follows:

$$H_{m_0} = 4 \cdot (m_0)^{1/2} \,. \tag{2}$$

On the other hand, the wave energy flux is a first indicator of the amount of wave energy available in a given area of the ocean. Wave energy flux P, or power density per meter of wave crest [33] can be computed as

$$P = \frac{\rho g^2}{4\pi} \int_0^{\infty} \frac{S(f)}{f} df = \frac{\rho g^2}{4\pi} m_{-1} = \frac{\rho g^2}{64\pi} H_{m_0}^2 \cdot T_e, \tag{3}$$

where ρ is the sea water density (1025 kg/m^3), g is the acceleration due to gravity, $H_{m_0} = 4\sqrt{m_0}$ is the spectral estimation of the significant wave height, and $T_e \equiv T_{-1,0} = m_{-1}/m_0$ is an estimation of the mean wave period, normally known as the period of energy, which is used in the design of turbines for wave energy conversion. Expression (3) (with H_{m_0} in meters and T_e in seconds) leads to

$$P = 0.49 \cdot H_{m_0}^2 \cdot T_e, \tag{4}$$

measured in kW/m, which helps engineers estimate the amount of wave energy available when planning the deployment of WECs at a given location.

3. The hybrid prediction system considered

In this paper we will optimize a hybrid prediction system for marine energy applications described in [25]. In this section we describe the main characteristics of this approach, in order to better explain later on the Bayesian optimization carried out on it. The prediction system is a hybrid wrapper approach, formed by a Grouping Genetic Algorithm for feature selection, and an Extreme Learning Machine to carry out the final prediction of H_{m_0} or P from a set of input data.

3.1. The Grouping Genetic Algorithm

The grouping genetic algorithm (GGA) [34, 35] is a type of evolutionary algorithm especially suited to tackle grouping problems, i.e., problems where a number of items must be assigned to a set of predefined groups. The GGA has shown very good performance on different real applications and problems [36, 37, 38, 39, 40, 41]. In the GGA, the encoding, crossover and mutation operators of traditional GAs are modified to better deal with grouping problems. In this paper we use the GGA to obtain a reduced set of features (feature selection) in a context of H_{m_0} and P prediction. We structure the description of the GGA in Encoding, Operators and Fitness Function calculation (Extreme Learning Machine).

3.1.1. Problem encoding

The GGA is a variable-length genetic algorithm. The encoding is defined by separating each individual in the algorithm into two parts: an *assignment* part, which associates each item to a given group, and a *group* part, which defines the groups that must be taken into account for the individual. In problems where the number of groups is not previously defined, it is straightforward that this is a variable-length algorithm: the group part varies from one individual to another. In our implementation of the GGA for feature selection, an individual $\mathbf{c}$ has the form $\mathbf{c} = [\mathbf{a}|\mathbf{g}]$. An example of an individual in the proposed GGA for a feature selection problem, with 20 features and 4 groups, is the following:

1 1 2 3 1 4 1 4 3 4 4 1 2 4 4 2 3 1 3 2 | 1 2 3 4

where the group 1 includes features $\{1, 2, 5, 7, 12, 18\}$, group 2 features $\{3, 13, 16, 20\}$, group 3 features {4,9,17,19} and finally group 4 includes features $\{6, 8, 10, 11, 14, 15\}$.

3.1.2. Genetic operators

In this paper we use a tournament-based selection mechanism, similar to the one described in [42]. This mechanism has been shown to be one of the most effective selection operators, avoiding super-individuals and performing a excellent exploration of the search space. Regarding the crossover operator, we have chosen a modified version of the one initially proposed by Falkenauer [34, 35]. It follows the process outlined in Figure 1:

1. Choose two parents from the current population, at random.

2. Randomly select two points for the crossover, from the "Groups" part of parent 1, then, all the groups between the two cross-points are selected. In the example of Figure 1 the two crossover points are G_1 and G_2. Note that, in this case the items of parent1 belonging to group G_1 and G_2 are 1, 2, 4, 5, and 6.

3. Insert the selected section of the "Groups" part into the second parent. After the insertion in the example of Figure 1, the assignment of the nodes 1, 2, 4, 5 and 6 of the offspring individual will be those of parent 1, while the rest of the nodes' assignment are those of parent 2. The "Groups" part of the offspring individual is that of parent 2 plus the selected section of parent 1 (8 groups in total, in this case).

4. Modify the "Groups" part of the offspring individual with their corresponding number. In the example, $G = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 1 \quad 2$ is modified into $G = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8$. Modify also the assignment part accordingly.

5. Remove any empty groups in the offspring individual. In the example considered, it is found that groups 1, 2, 3, and 6 are empty, so we can eliminate these groups identification number and rearrange the rest. The final offspring is then obtained.

Regarding mutation operator, we apply a swapping mutation in which two items are interchanged (swapping this way the assignment of features to different groups). This procedure is carried out with a very low probability ($P_m = 0.01$), to avoid increasing of the random search in the process. In the next section we describe the fitness function used to guide the search in the GGA, the ELM neural network, which is a very fast algorithm with excellent performance in prediction problems.

3.1.3. Fitness function: the Extreme Learning Machine

An ELM [43] is a fast learning method based on the structure of MLPs with a novel way of training feed-forward neural networks. One of the most important characteristics of the ELM training is the randomness in the process where the network weights are set, obtaining, in this way, a pseudo-inverse of the hidden-layer output matrix. The simplicity of this technique makes the training algorithm extremely fast. Moreover, it is remarkable the outstanding performance shown when compared to other learning methods. For example, it usually outperforms other established approaches such as classical MLPs or SVRs [43]. ELMs have recently been used within hybrid wrapper systems for feature selection [44, 45], similarly as we use them in this paper.

The ELM algorithm can be explained as follows: given a training set

$$\mathbb{T} = (\mathbf{x}_i, \boldsymbol{W}_i)|\mathbf{x}_i \in \mathbb{R}^n, \boldsymbol{W}_i \in \mathbb{R}, i = 1, \cdots, l\,,$$

an activation function $g(x)$ and number of hidden nodes ($\tilde{N}$),

1. Randomly assign inputs weights $\mathbf{w}_i$ and bias b_i, $i = 1, \cdots, \tilde{N}$.
2. Calculate the hidden layer output matrix $\mathbf{H}$, defined as

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1\mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}\mathbf{x}_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1\mathbf{x}_l + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}\mathbf{x}_N + b_{\tilde{N}}) \end{bmatrix}_{l \times \tilde{N}} . \tag{5}$$

3. Calculate the output weight vector β as

$$\beta = \mathbf{H}^{\dagger}\mathbf{T}, \tag{6}$$

   where $\mathbf{H}^{\dagger}$ stands for the Moore-Penrose inverse of matrix $\mathbf{H}$ [43], and $\mathbf{T}$ is the training output vector, $\mathbf{T} = [\boldsymbol{W}_1, \cdots, \boldsymbol{W}_l]^T$.

The number of hidden nodes ($\tilde{N}$) is a free parameter of the ELM training, and it can be fixed initially, or in a best convenient way, it must be estimated for obtaining good results as a part of a validation set in the learning process. Hence, scanning a range of $\tilde{N}$ values is the solution for this problem.

The Matlab ELM implementation by G. B. Huang, freely available on the Internet [46], has been used in this paper.

4. Bayesian optimization of the prediction system

Every machine learning algorithm or prediction system has its own set of parameters that must be adjusted to obtain an optimal performance. An example is a deep neural network in which one has to specify parameters such as the learning rate, the number of layers, the number of neurons in each layer, etc. [47]. Another example is stochastic gradient boosting in which one has to choose the number of terminal nodes in the ensemble trees, the number of trees, the regularization parameter, etc. [48]. In our particular setting, in an ELM the number of units in the hidden layer has to be specified before training; and in the genetic algorithm described in Section 3.1, the probability of mutation and the number of epochs must be known initially.

Changing the parameter values of a prediction system may have a strong impact in its performance. Parameter tuning is hence defined as the problem of finding the optimal parameter values of a prediction system on the problem considered. This task has traditionally been addressed by human experts, which often use prior knowledge to specify parameter values that are expected to perform well. However, such an approach can suffer from human bias. An alternative solution is to consider a grid or uniform search in the space of parameters to look for values that result in a good performance on a validation set. These methods, however, suffer when the dimensionality of the parameter space is very high [49]. In those settings they often require a large number of parameter evaluations.

Bayesian optimization (BO) has emerged as practical tool for parameter selection in prediction systems. These methods provide an efficient alternative to a grid or uniform search of the parameter space [29]. Assume that the surface defined by the error of a prediction system that depends on some parameters is smooth. In that case, we can search through the parameter space according to a criterion that exploits this smoothness property and avoids exhaustive exploration. More precisely, BO methods are very useful for optimizing black-box objective functions that lack an analytical expression (which means no gradient information), are very expensive to evaluate, and in which the evaluations are potentially noisy [50, 51, 52]. The performance of a prediction system on a randomly chosen validation set, when seen as a function of the chosen parameters, has all these characteristics.

Consider a black-box objective $f(\cdot)$ with noisy evaluations of the form $y_i = f(\mathbf{x}_i) + \epsilon_i$, with ϵ_i some noise term. BO methods are very successful at reducing the number of evaluations of the objective function needed to solve the optimization problem. At each iteration $t = 1, 2, 3, \ldots$ of the optimization process, these methods fit a probabilistic model, typically a Gaussian process (GP) to the observations of objective function $\{y_i\}_{i=1}^{t-1}$ collected so far. The uncertainty about the objective function provided by the GP is then used to generate an acquisition function $\alpha(\cdot)$, whose value at each input location indicates the expected utility of evaluating $f(\cdot)$ there. The next point $\mathbf{x}_t$ at which to evaluate the objective $f(\cdot)$ is the one that maximizes $\alpha(\cdot)$. Importantly, $\alpha(\cdot)$ only depends on the probabilistic model and can hence be evaluated with very little cost. Thus, this function can be maximized very quickly using standard optimization techniques. This process is repeated until enough data about the objective has been collected. When this is the case, the GP predictive mean for $f(\cdot)$ can be optimized to find the solution of the optimization problem. Algorithm 1 shows the details of such a process.

The key for BO success is that evaluating the acquisition function $\alpha(\cdot)$ is very cheap compared to the evaluation of the actual objective $f(\cdot)$, which in our case requires re-training the prediction system. This is so because the

for $t = 1, 2, 3, \ldots, max\_steps$ **do**

1: Find the next point to evaluate by optimizing the acquisition function: $\mathbf{x}_t = \arg\max_{\mathbf{x}} \quad \alpha(\mathbf{x}|\mathcal{D}_{1:t-1})$.

2: Evaluate the black-box objective $f(\cdot)$ at $\mathbf{x}_t$: $y_t = f(\mathbf{x}_\mathrm{t}) + \epsilon_t$.

3: Augment the observed data $\mathcal{D}_{1:t} = \mathcal{D}_{1:t-1} \bigcup \{\mathbf{x}_t, y_t\}$.

4: Update the Gaussian process model using $\mathcal{D}_{1:t}$.

end

Result: Optimize the mean of the Gaussian process to find the solution.

Algorithm 1: Bayesian optimization of a black-box objective function.

acquisition function only depends on the GP predictive distribution for $f(\cdot)$ at a candidate point $\mathbf{x}$. Let the observed data until step $t-1$ of the algorithm be $\mathcal{D}_i = \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1}$. The GP predictive distribution for $f(\cdot)$ is given by a Gaussian distribution characterized by a mean $\mu(\mathbf{x})$ and a variance $\sigma^2(\mathbf{x})$. These values are:

$$\mu(\mathbf{x}) = \boldsymbol{k}_*^T(\mathbf{K} + \sigma_n^2 I)^{-1}\boldsymbol{y}\,, \tag{7}$$

$$\sigma^2(\mathbf{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_*^T(\mathbf{K} + \sigma_n^2 I)^{-1}\boldsymbol{k}_*\,. \tag{8}$$

where $y = (y_1, \ldots, y_{t-1})$ is a vector with the objective values observed so far; $\mathbf{k}_*$ is a vector with the prior covariances between $f(\mathbf{x})$ and each y_i; $\mathbf{K}$ is a matrix with the prior covariances among each y_i, for $i = 1, \ldots, t-1$; and $k(\boldsymbol{x}, \boldsymbol{x})$ is the prior variance at the candidate location $\mathbf{x}$. All these quantities are obtained from a covariance function $k(\cdot, \cdot)$ which is pre-specified and receives as an input two points, $\mathbf{x}_i$ and $\mathbf{x}_j$, at which the covariance between $f(\mathbf{x}_i)$ and $f(\mathbf{x}_j)$ has to be evaluated. A typical covariance function employed for BO is the Matérn function [29]. For further details about GPs we refer the reader to [53].

Thus, BO methods typically spend a little bit of time thinking very carefully where to evaluate next the objective function with the aim of finding its optimum with the smallest number of evaluations. This is a very useful strategy when the objective function is very expensive to evaluate and it can save a lot of computational time. Three steps of the BO optimization process are illustrated graphically in Fig. 2 for a toy minimization problem.

Unlike BO methods, grid or uniform search strategies are based in a pure exploration of the search space. If we make the assumption that the objective function is smooth, doing a few evaluations in regions of the input space that look more promising (exploitation) is expected to give better results. In BO methods the acquisition function $\alpha(\cdot)$ balances between exploration and exploitation in an automatic way. An example of an acquisition function is expected improvement (EI) [54]. EI is obtained as the expected value under the GP predictive distribution for y_i, of the utility function $u(y_i) = \max(0, \nu - y_i)$, where $\nu = \min(\{y_i\}_{i=1}^{t-1})$ is the best value observed so far. That is, EI measures on average how much we will improve on the current best solution by evaluating the objective at each candidate point. An advantage of EI is that the corresponding acquisition function $\alpha(\cdot)$ can be computed analytically: $\alpha(\mathbf{x}) = \sigma(\mathbf{x})(\gamma(\mathbf{x})\Phi(\gamma(\mathbf{x}) + \phi(\gamma(\mathbf{x}))$, where $\gamma(\mathbf{x}) = (\nu - \mu(\mathbf{x}))/\sigma(\mathbf{x})$ and $\Phi(\cdot)$ and $\phi(\cdot)$ are respectively the c.d.f. and p.d.f. of a standard Gaussian. EI is the acquisition function displayed in Fig. 2.

BO has been recently applied with success in different prediction systems for finding good parameter values. For example, it has been used to find the parameters of topic models based on latent Dirichlet allocation, support vector machines, or deep convolutional neural networks [29]. Furthermore, BO methods have also been used to optimize a logistic regression model for labelling Amazon product reviews [55], or to optimize the weights of a neural network to balance vertical poles and lengths on a moving cart [56]. Another applications of BO are found in the field of environmental monitoring, in the task of adjusting the parameters of a control system for robotics, in the optimization of recommender systems, and in combinatorial optimization [51, 52]. Finally, BO methods has been implemented in different software packages. An implementation in python is called Spearmint and is available at [57], which is the BO implementation used in this work.

5. Experiments and results

This section describes some experiments with the aim of showing the improvements obtained in the performance of the prediction system when its parameters are optimized with the Bayesian techniques introduced before. We consider a real problem of wave energy flux prediction ($P = 0.49 \cdot H_s^2 \cdot T_e$ kW/m, [31]) from marine buoys. Figure 5 shows the three buoys considered in this study at the Western coast of the USA, whose data bases are obtained from [58]. The objective of the problem is to carry out the reconstruction of buoy 46069 from a number of predictive variables from the other two buoys. Thus, 10 predictive variables measured at each neighbor buoy are considered (a total of 20 predictive variables to carry out the reconstruction). Table 1 shows details of the predictive variables for this problem. Data for two complete years (1st January 2009 to 31st December 2010) are used, since complete data (without missing values in predictive and objective P) are available for that period in the three buoys. These data are divided into training set (year 2009) and test set (year 2010) to evaluate the performance of the proposed algorithm.

We have divided this experimental section into two different subsections. First, we show the performance of the BO techniques proposed in the optimization of the specific GGA-ELM prediction algorithm. Second, we will show how the prediction performance is improved when the system is run with the parameters obtained by the BO techniques, i.e. by comparing the performance of the system before and after tuning the parameters with BO.

5.1. Bayesian optimization of the wave energy prediction system parameters

We evaluate the utility of the BO techniques described in Section 4 for finding good parameters for the prediction system described in Section 3. More precisely, we try to find the parameters that minimize the RMSE of the best individual found by the GGA on a validation set that contains 33% of the total data available. The parameters of the GGA that are adjusted are the probability of mutation $p \in [0, 0.3]$, the percentage of confrontation in the tournament $q \in [0.5, 1.0]$, and the number of epochs $e \in [50, 200]$. On the other hand, the parameters of the ELM that is used to evaluate the fitness in the GGA are also adjusted. These parameters are the number of hidden units $n \in [50, 150]$ and the logarithm of the regularization constant of a ridge regression estimator, that is used to find the weights of the output layer $\gamma \in [-15, -3]$. Note that a ridge regression estimator for the output layer weights allows for a more flexible model than the standard ELM, as the standard ELM is retrieved when γ is negative and large [59].

We compare the BO method with two techniques. The first technique is a random exploration of the space of parameters. The second technique is a configuration specified by a human expert. Namely, $p = 0.02$, $q = 0.8$, $e = 200$, $n = 150$ and $\gamma = -10$. These are reasonable values that are expected to perform well in the specific application tackled. We set our computational budget to 50 different parameter evaluations for both the BO and the random exploration strategy. After each evaluation, we report the performance of the best solution found. The experiments are repeated for 50 different random seeds and we report average results. All BO experiments are carried out using the acquisition function EI and the software for BO Spearmint.

Fig. 3 and 4 show the average results obtained and the corresponding error bars for the task of predicting the wave energy flux and the task of predicting the wave height, respectively. Each figure shows the average RMSE of each method (BO and random exploration) on the validation set as a function of the number of configurations evaluated. The performance of the configuration specified by a human expert is also shown. We observe that the BO strategy performs best in each setting. In particular, after a few evaluations the BO method is able to outperform the results of the human expert and it provides results that are similar or better than the ones obtained by the random exploration strategy with a smaller number of evaluations.

5.2. Estimation of the generalization performance

In a second round of experiments, we show the performance of the proposed prediction system after its optimization with the BO methodology. Note that after the feature selection process with the GGA-ELM approach, we use an ELM and a SVR [60, 61] to obtain the final prediction of the wave energy flux P and significant wave height H_s.

Table 2 shows the results obtained for the experiments carried out. We can observe the comparison between ELM and SVR approaches in different scenarios: the prediction obtained with all the features, the prediction obtained with the hybrid algorithm GGA-ELM (without BO methodology), and finally the prediction acquired after the application of the BO process in the GGA-ELM approach. As Table 2 summarizes, we can see how the hybrid GGA-ELM algorithm improves the results obtained by the ELM and SVR approaches (without feature selection). In fact, the SVR algorithm improves the values of the Pearson's Correlation Coefficient (r^2) around 75% in the case of the feature selection method, against the poor 31% when all features are used. Moreover, these results are improved by means of the BO methodology, using ELM and SVR approaches after the GGA-ELM. In the case of the ELM, we get values of the r^2 around 77% against the 71% achieved with the GGA-ELM algorithm without the BO improvement. The same behavior is obtained for the SVR algorithm: we have values around 78%

with the application of the BO methodology against the 75% obtained for the GGA-ELM approach when the parameters are fixed by a human expert.

The results of the previous tables can be better visualized in the following graphics. In Fig. 6 the temporary predictions carried out by the ELM and SVR approaches are shown. We can see how the cases (c) and (d) improve the approximation to the real values against the cases (a) and (b) where the BO methodology is not applied. The same situation can be seen in Fig. 8, where the scatter plots are presented for the results obtained with and without the BO methodology.

The same procedure is carried out in the case of the H_s. Table 3 compares the results obtained in the different experiments. As it can be seen, the results are improved with the use of the BO methodology with values of the r^2 around 74% for the ELM and SVR predictions, against the 66% and 39% achieved for the ELM and SVR, respectively, with all features. The GGA-ELM algorithm improves these last results, but they are not so good like when we use the BO methodology. In Fig. 7 the temporary predictions for the GGA-ELM-ELM, GGA-ELM-SVR, BO-GGA-ELM-ELM and BO-GGA-ELM-SVR are shown. The same is done for the scatter plots, whose Fig. 9, present the results mentioned above.

In both predictions (P and H_s) the BO methodology improves the results, for this reason we can highlight the generality of the method.

6. Conclusions

In this paper we have shown how a hybrid prediction system for wave energy prediction can be improved by means of Bayesian optimization (BO) methodology. The prediction system is formed by a grouping genetic algorithm for feature selection, and an Extreme Learning Machine for effective prediction of the target variable, the wave energy flux in this case. After this feature selection process, the final prediction of the wave energy flux is obtained by means of an ELM or a SVR approach. The paper describes in detail the BO methodology, and its specific application in the optimization of the GGA-ELM for a real problem of wave energy flux prediction from buoys data in Western California USA. The results show that the BO methodology is able to improve the performance of the systems, i.e., the prediction of the optimized systems is significantly better than that of the system without the BO methodology applied.

Acknowledgements

This work has been partially supported by *Comunidad de Madrid*, under projects number S2013/ICE-2933 and S2013/ICE-2845, and by National projects TIN2014-54583-C2-2-R, TIN2013-42351-P and TIN2016-76406-P of the *Spanish Ministerial Commission of Science and Technology (MICYT).* We acknowledge support by DAMA network TIN2015-70308-REDT. We acknowledge the use of the facilities of Centro de Computación Científica de la UAM.

References

[1] Z. Zheng and L. Sun, "Path following control for marine surface vessel with uncertainties and input saturation," Neurocomputing, vol. 177, pp. 158-167, 2016.

[2] L. Liu, D. Wang and Z. Peng, "Path following of marine surface vehicles with dynamical uncertainty and time-varying ocean disturbances," Neurocomputing, vol. 173, pp. 799-808, 2016.

[3] F. Comola, T. Lykke Andersen, L. Martinelli, H.F. Burcharth and P. Ruol, "Damage pattern and damage progression on breakwater roundheads under multidirectional waves," Coastal Engineering, vol. 83, pp. 24-35, 2014.

[4] S.W. Kim and K.D. Suh, "Determining the stability of vertical breakwaters against sliding based on individual sliding distances during a storm," Coastal Engineering, vol. 94, pp. 90-101, 2014.

[5] R.A. Arinaga, K.F. Cheung, "Atlas of global wave energy from 10 years of reanalysis and hindcast data," Renewable Energy, vol. 39, pp. 49-64, 2012.

[6] M. Esteban, D. Leary, "Current developments and future prospects of offshore wind and ocean energy," Applied Energy, vol. 90, pp. 128-136, 2012.

[7] I. López, J. Andreu, S. Ceballos, I. Martínez de Alegría and I. Kortabarria, "Review of wave energy technologies and the necessary power-equipment," Renewable and Sustainable Energy Reviews, vol. 27, pp. 413-434, 2013.

[8] S. Rao and S.Mandal,2005. “Hindcasting of storm waves using neural networks,” Ocean Engineering, vol. 32, pp.667-684, 2005.

[9] A.S. Bahaj, “Generating electricity from the oceans,” Renewable and Sustainable Energy Reviews, vol. 15, pp. 3399-3416, 2011.

[10] A.F. Falcão, “Wave energy utilization: A review of the technologies,” Renewable and Sustainable Energy Reviews, vol. 14, pp. 899-918, 2010.

[11] M. Fadaeenejad, R. Shamsipour, S.D. Rokni, C. Gomes, “New approaches in harnessing wave energy: With special attention to small islands,” Renewable and Sustainable Energy Reviews, vol. 29, pp. 345-354, 2014.

[12] L. Rusu, C. Guedes-Soares, “Wave energy assessments in the Azores islands,” Renewable Energy, vol. 45, pp. 183-196, 2012.

[13] L. Cuadra, S. Salcedo-Sanz, J. C. Nieto-Borge, E. Alexandre and G. RodrÃguez, “Computational Intelligence in wave energy: comprehensive review and case study,” Renewble and Sustainable Energy Reviews, vol. 58, pp. 1223-1246, 2016.

[14] M.C. Deo and C.S. Naidu, “Real time wave prediction using neural networks,” Ocean Engineering, vol. 26(3), pp. 191-203, 1998.

[15] J.D. Agrawal and M.C. Deo, “Wave parameter estimation using neural networks,” Marine Structures, vol. 17, pp. 536-550, 2004.

[16] C.P. Tsai, C. Lin and J.N. Shen, “Neural network for wave forecasting among multi-stations,” Ocean Engineering, vol. 29(13), pp. 1683-1695, 2002.

[17] B. Robertson, B. Gharabaghi and K. Hall, “Prediction of incipient breaking wave heights using artificial neural networks and empirical relationships,” Coastal Engineering Journal, vol. 57(4), 1550018, pp. 1-27, 2015.

[18] A. Castro, R. Carballo, G. Iglesias and J.R. Rabuñal, “Performance of artificial neural networks in nearshore wave power prediction,” Applied Soft Computing, vol. 23, pp. 194-201, 2014.

[19] M. Zanaganeh, S. Jamshid-Mousavi and A.F. Etemad-Shahidi, "A hybrid genetic algorithm-adaptive network-based fuzzy inference system in prediction of wave parameters," Engineering Applications of Artificial Intelligence, vol. 22(8), pp. 1194-1202, 2009.

[20] E. Alexandre, L. Cuadra, JC. Nieto-Borge, G. Candil-García, M. del Pino and S. Salcedo-Sanz, "A Hybrid Genetic Algorithm – Extreme Learning Machine approach for accurate significant wave height reconstruction," Ocean Modelling, vol. 92, pp. 115-123, 2015.

[21] L. Cornejo-Bueno, J. C. Nieto-Borge, P. García-Díaz, G. Rodríguez and S. Salcedo-Sanz, "Significant Wave Height and Energy Flux Prediction for Marine Energy Applications: A Grouping Genetic Algorithm – Extreme Learning Machine Approach," Renewable Energy, vol. 97, pp. 380-389, 2016.

[22] J. Mahjoobi, A. Etemad-Shahidi and M.H. Kazeminezhad, "Hindcasting of wave parameters using different soft computing methods," Applied Ocean Research, vol. 30(1), pp. 28-36, 2008.

[23] J. Mahjoobi and E.A. Mosabbeb, "Prediction of significant wave height using regressive support vector machines," Ocean Engineering, vol. 36(5), pp. 339-347, 2009.

[24] S. Salcedo-Sanz, J. C. Nieto-Borge, L. Carro-Calvo, L. Cuadra, K. Hessner and E. Alexandre, "Significant wave height estimation using SVR algorithms and shadowing information from simulated and real measured X-band radar images of the sea surface," Ocean Engineering, vol. 101, pp. 244-253, 2015.

[25] L. Cornejo-Bueno, J. C. Nieto-Borge, E. Alexandre, K. Hessner and S. Salcedo-Sanz, "Accurate estimation of significant wave height with support vector regression algorithms and marine radar images," Coastal Engineering, vol. 114, pp. 233-243, 2016.

[26] J.C. Fernández, S. Salcedo-Sanz, P.A. Gutiérrez, E. Alexandre and C. Hervás-Martínez, "Significant wave height and energy flux range forecast with machine learning classifiers," *Engineering Applications of Artificial Intelligence*, vol. 43, pp. 44-53, 2015.

[27] S.P. Nitsure, S.N. Londhe and K.C. Khare, "Wave forecasts using wind information and genetic programming," Ocean Engineering, vol. 54, pp. 61-69, 2012.

[28] M.Özger, "Prediction of ocean wave energy from meteorological variables by fuzzy logic modeling," Expert Systems with Applications, vol. 38(5), pp. 6269-6274, 2011.

[29] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms." Advances in neural information processing systems, 2012.

[30] J.C. Nieto-Borge, K. Reichert and K. Hessner, "Detection of spatio-temporal wave grouping properties by using temporal sequences of X-band radar images of the sea surface," Ocean Modelling, vol. 61, pp. 21-37, 2013.

[31] Y. Goda, "Random seas and design of maritime structures," World Scientific, 2010.

[32] http://www.ndbc.noaa.gov/: National Data Buoy Center. National Oceanic and Atmospheric Administration of the USA (NOAA) (accessed June 29, 2016).

[33] B.G. Cahill and T. Lewis, "Wave energy resource characterization of the Atlantic marine energy test site," International Journal of Marine Energy, vol. 1, pp. 3-15, 2013.

[34] E. Falkenauer, "The grouping genetic algorithm–widening the scope of the GAs, Belgian journal of operations research," Statistics and Computer Science, vol. 33, pp. 79-102, 1992.

[35] E. Falkenauer, "Genetic algorithms for grouping problems," New York: Wiley, 1998.

[36] L.E. Agustín-Blas, S. Salcedo-Sanz, P. Vidales, G. Urueta and J.A. Portilla-Figueras, "Near optimal citywide WiFi network deployment using a hybrid grouping genetic algorithm," Expert Systems with Applications, vol. 38(8), pp. 9543-9556, 2011.

[37] L.E. Agustín-Blas, S. Salcedo-Sanz, E.G. Ortiz-García, J.A. Portilla-Figueras and A.M. Pérez-Bellido, "A hybrid grouping genetic algorithm for assigning students to preferred laboratory groups," Expert Systems with Applications, vol. 36, pp. 7234-7241, 2009.

[38] E.C. Brown and R.T. Sumichrast, "Evaluating performance advantages of grouping genetic algorithms," Engineering Applications of Artificial Intelligence, vol. 18(1), pp. 1-12, 2005.

[39] T. James, E.C. Brown and K.B. Keeling, "A hybrid grouping genetic algorithm for the cell formation problem," Computers & Operations Research, vol. 34, pp. 2059-2079, 2007.

[40] T. James, M. Vroblefski and Q. Nottingham, "A hybrid grouping genetic algorithm for the registration area planning problem," Computer Communications, vol. 30(10), pp. 2180-2190, 2007.

[41] P. De Lit, E. Falkenauer and A. Delchambre, "Grouping genetic algorithms: an efficient method to solve the cell formation problem," Mathematics and Computers in Simulation, vol. 51(3-4), pp. 257-271, 2000.

[42] X. Yao, Y. Liu and G. Lin, "Evolutionary Programming made faster," IEEE Transactions on Evolutionary Computation, vol. 3(2), pp. 82-102, 1999.

[43] G.B. Huang and Q.Y. Zhu, "Extreme learning machine: theory and applications," Neurocomputing, vol. 70, pp. 489-501, 2006.

[44] E. Alexandre, L. Cuadra, S. Salcedo-Sanz, A. Pastor-Sánchez and C. Casanova-Mateo, "Hybridizing extreme learning machines and genetic algorithms to select acoustic features in vehicle classification applications," Neurocomputing, vol. 152, pp. 58-68, 2015.

[45] S. Salcedo-Sanz, A Pastor-Sánchez, L Prieto, A Blanco-Aguilera and R. García-Herrera, "Feature selection in wind speed prediction systems based on a hybrid coral reefs optimization-Extreme learning machine approach," Energy Conversion and Management, vol. 87, pp. 10-18, 2014.

[46] G. B. Huang, "ELM matlab code," `http://www.ntu.edu.sg/home/egbhuang/elm_codes.html`

[47] Y. LeCun, B. Yoshua and G. Hinton, "Deep learning," Nature, vol. 521.7553, pp. 436-444, 2015.

[48] J. H. Friedman, "Stochastic gradient boosting," Computational Statistics & Data Analysis vol. 38.4, pp. 367-378, 2002.

[49] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," Journal of Machine Learning Research, vol. 13, pp. 281-305, 2012.

[50] J. Mockus, V. Tiesis and A. Zilinskas. "The application of Bayesian methods for seeking the extremum," Towards Global Optimization, vol. 2, pp. 117-129, 1978.

[51] E. Brochu, V. M. Cora, and N. De Freitas. "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," arXiv preprint arXiv, pp. 1012-2599, 2010.

[52] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. "Taking the human out of the loop: A review of Bayesian optimization," Proceedings of the IEEE, vol. 104, pp. 148-175, 2016.

[53] C. E. Rasmussen. "Gaussian processes for machine learning," 2006.

[54] D. R. Jones, M. Schonlau, and W. J. Welch. "Efficient global optimization of expensive black-box functions," Journal of Global Optimization, vol. 13(4), pp. 455-492, 1998.

[55] I. Dewancker, M. McCourt and S. Clark, "Bayesian Optimization for Machine Learning: A Practical Guidebook," arXiv preprint arXiv, vol. 1612.04858, 2016.

[56] M. Frean and P. Boyle, "Using Gaussian processes to optimize expensive functions," In W. Wobcke and M. Zhang, editors, AI: Advances in Artifcial Intelligence, vol. 5360 of Lecture Notes in Computer Science, pp. 258-267. Springer Berlin / Heidelberg, 2008.

[57] BO method implementation (Python): `https://github.com/HIPS/Spearmint`.

[58] NOAA, National Data Buoy Center: http://www.ndbc.noaa.gov/. Last accessed 5th May 2016.

[59] A. Albert. "Regression and the Moore-Penrose pseudoinverse," (No. 519.536 A5), 1972.

[60] A.J. Smola and B. Schölkopf, "A tutorial on support vector regression," Statistics and Computing, vol. 14, pp. 199-222, 2004.

[61] S. Salcedo-Sanz, J.L. Rojo, M. Martínez-Ramón and G. Camps-Valls, "Support vector machines in engineering: an overview," WIREs Data Mining and Knowledge Discovery, vol. 4(3), pp. 234-267, 2014.

[62] David H. Wolpert, and G. M. William, "No free lunch theorems for optimization," IEEE transactions on evolutionary computation vol. 1.1, pp. 67-82, 1997.

[63] E. Vazquez and J. Bect, "Convergence properties of the expected improvement algorithm with fixed mean and covariance functions," Journal of Statistical Planning and inference, vol. 140.11, pp. 3088-3095, 2010.

[64] C. Zhu et al, "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization," ACM Transactions on Mathematical Software (TOMS), vol. 23.4, pp. 550-560, 1997.

List of Tables

List of Figures

Table 1: Predictive variables used in the experiments.

Acronym	Predictive variable	units
WDIR	Wind direction	[degrees]
WSPD	Wind speed	[m/s]
GST	Gust speed	[m/s]
WVHT	Significant wave height	[m]
DPD	Dominant wave period	[sec]
APD	Average period	[sec]
MWD	Direction DPD	[degrees]
PRES	Atmospheric pressure	[hPa]
ATMP	Air temperature	[Celsius]
WTMP	water temperature	[Celsius]

Table 2: Comparative results of the P estimation by the ELM and SVR approaches after the feature selection by the GGA-ELM in 2010.

Experiments	RMSE	MAE	r^2
All features-ELM	3.4183 kW/m	2.4265 kW/m	0.6243
All features-SVR	4.4419 kW/m	2.8993 kW/m	0.3129
GGA-ELM-ELM	2.8739 kW/m	1.8715 kW/m	0.7101
GGA-ELM-SVR	2.6626 kW/m	1.6941 kW/m	0.7548
BO-GGA-ELM-ELM	2.5672 kW/m	1.7596 kW/m	0.7722
BO-GGA-ELM-SVR	2.4892 kW/m	1.6589 kW/m	0.7823

Table 3: Comparative results of the H_s estimation by the ELM and SVR approaches after the feature selection by the GGA-ELM in 2010.

Experiments	RMSE	MAE	r^2
All features-ELM	0.4653 m	0.3582 m	0.6624
All features-SVR	0.6519 m	0.4986 m	0.3949
GGA-ELM-ELM	0.3650 m	0.2858 m	0.7049
GGA-ELM-SVR	0.3599 m	0.2727 m	0.7056
BO-GGA-ELM-ELM	0.3324 m	0.2519 m	0.7429
BO-GGA-ELM-SVR	0.3331 m	0.2461 m	0.7396

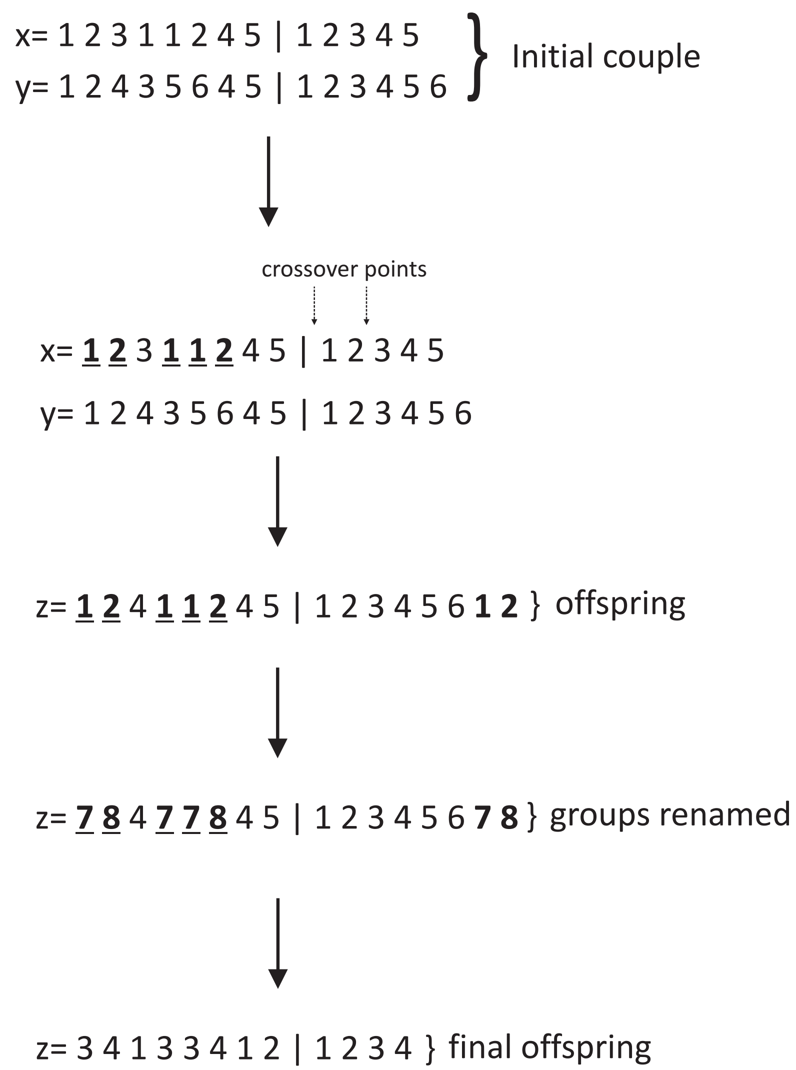

Figure 1: Outline of the grouping crossover implemented in the proposed GGA.

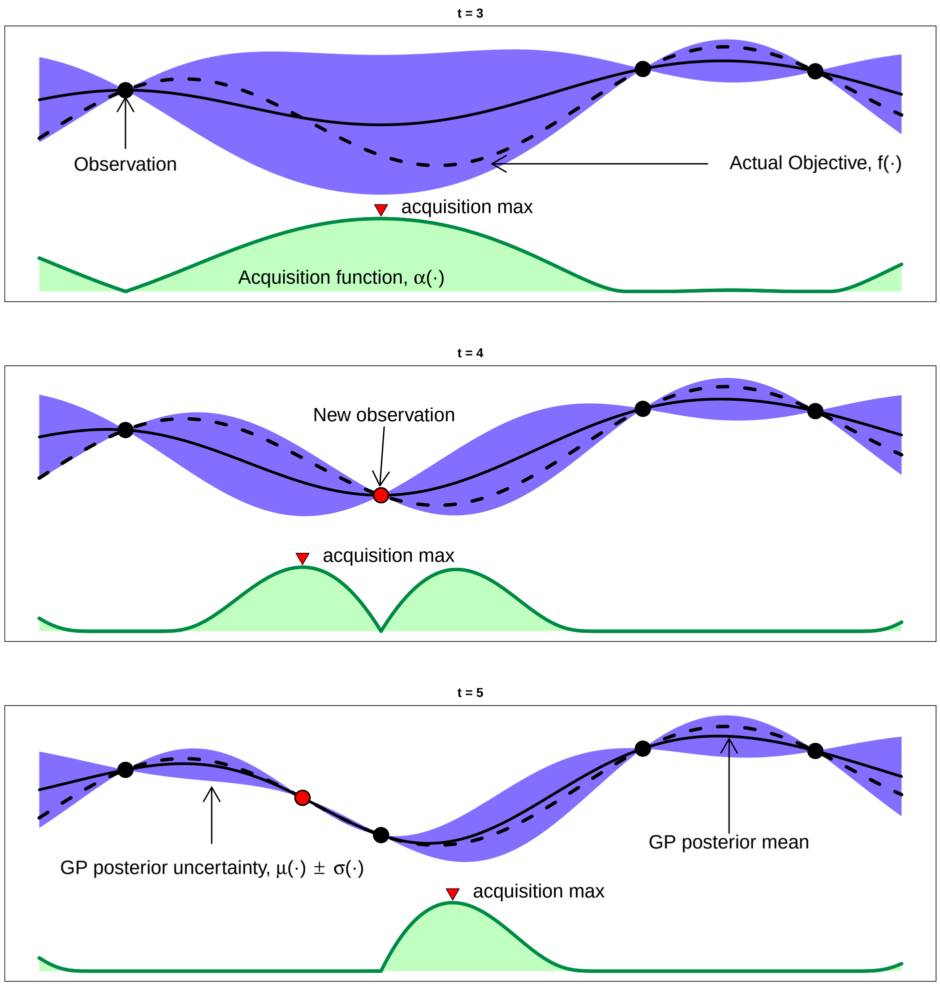

Figure 2: An example of BO on a toy 1D noiseless problem. The figures show a GP estimation of the objective $f(\cdot)$ over three iterations. The acquisition function is shown in the lower part of the plot. The acquisition is high where the GP predicts a low objective and where the uncertainty is high. Those regions in which it is unlikely to find the global minimum of $f(\cdot)$ have low acquisition values and will not be explored.

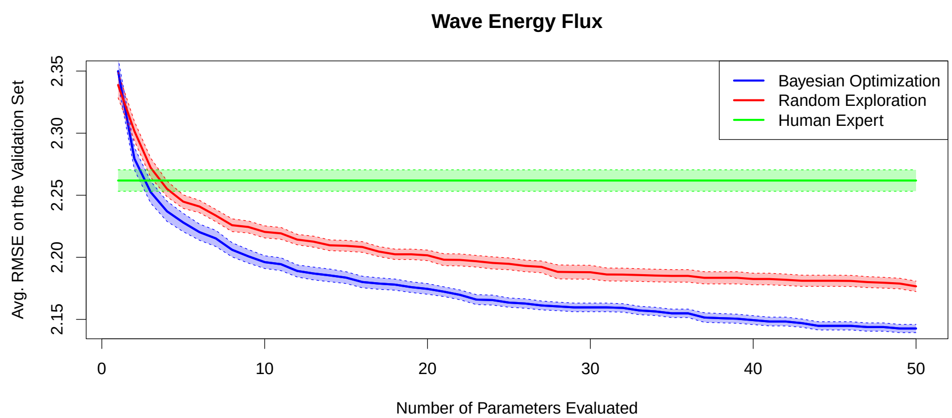

Figure 3: Average results obtained for the Wave Energy Flux optimization after evaluating the performance of 50 different parameters for the BO technique and a random exploration of the parameter space. The performance a configuration specified by a human expert is also shown for comparison.

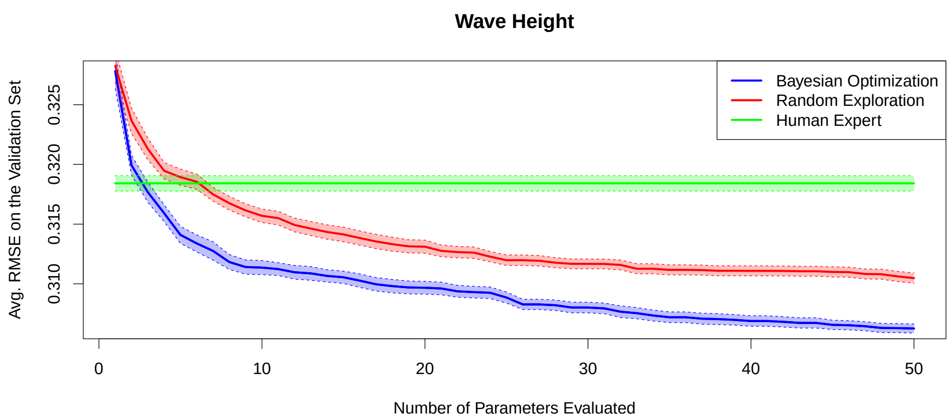

Figure 4: Wave Height optimization average results of the performance of the 50 different parameter values selected by the BO technique and a random exploration of the parameter space. The plot also shows the performance of the parameter values selected by a human expert.

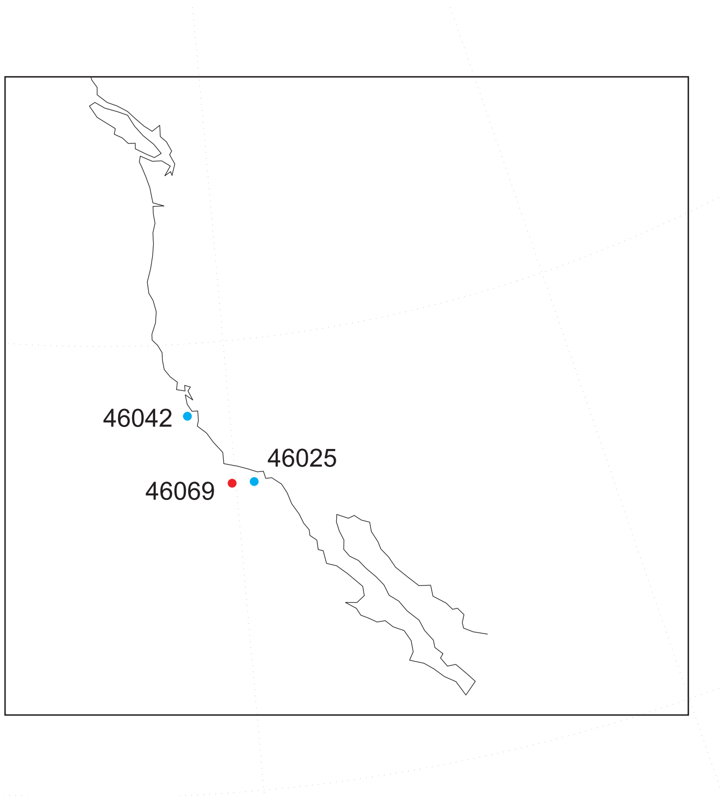

Figure 5: Western USA Buoys considered in this study. In red buoy where the P prediction is carried out from data at blue ones.

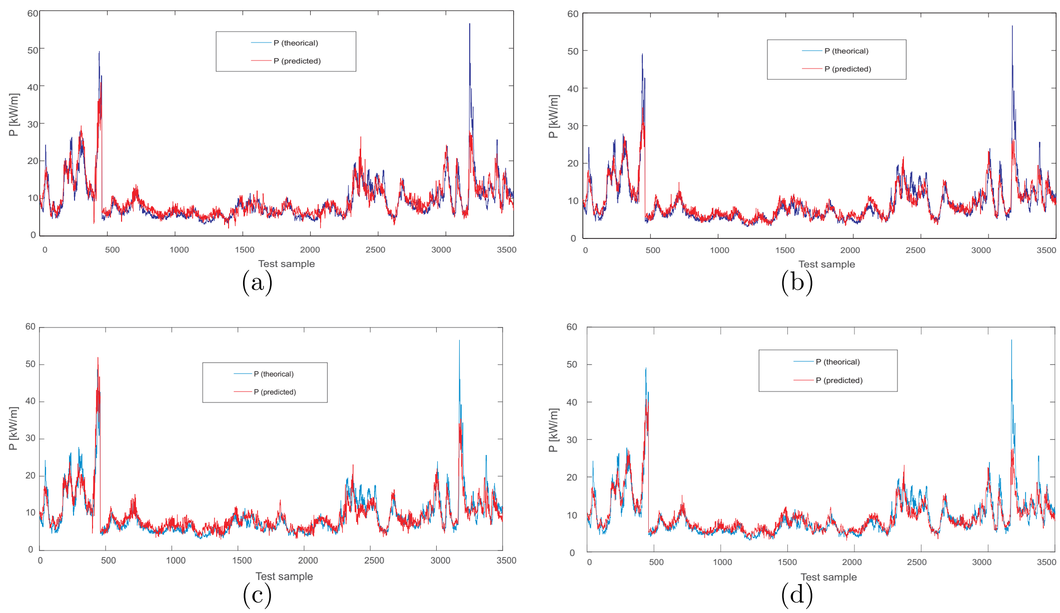

Figure 6: P prediction after the feature selection process with the GGA-ELM approach; (a) ELM; (b) SVR; (c) ELM with Bayesian optimization; (d) SVR with Bayesian optimization.

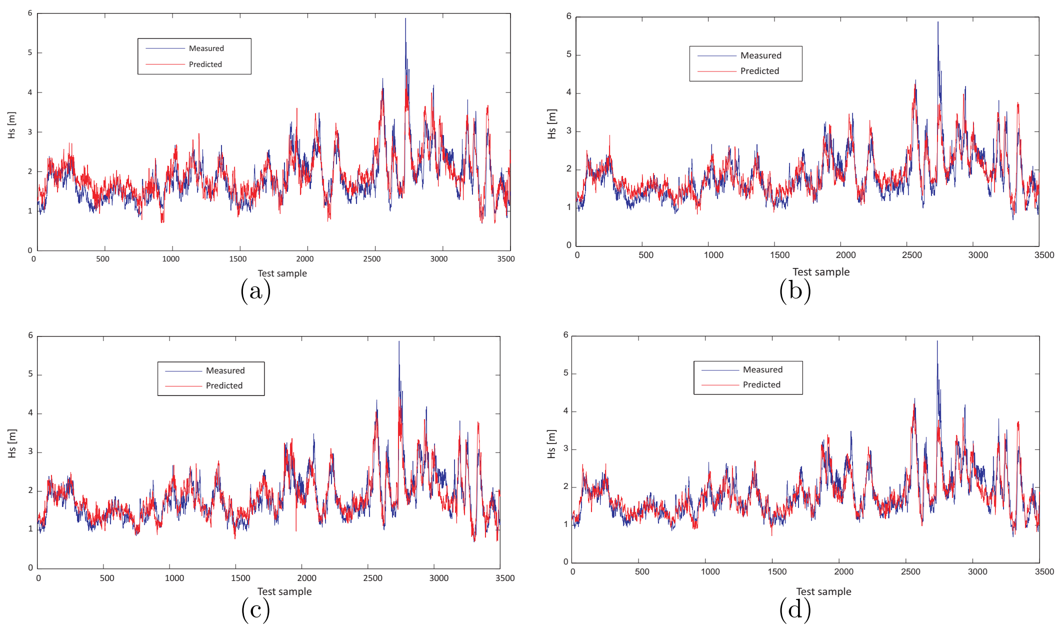

Figure 7: H_s prediction after the feature selection process with the GGA-ELM approach; (a) ELM; (b) SVR; (c) ELM with Bayesian optimization; (d) SVR with Bayesian optimization.

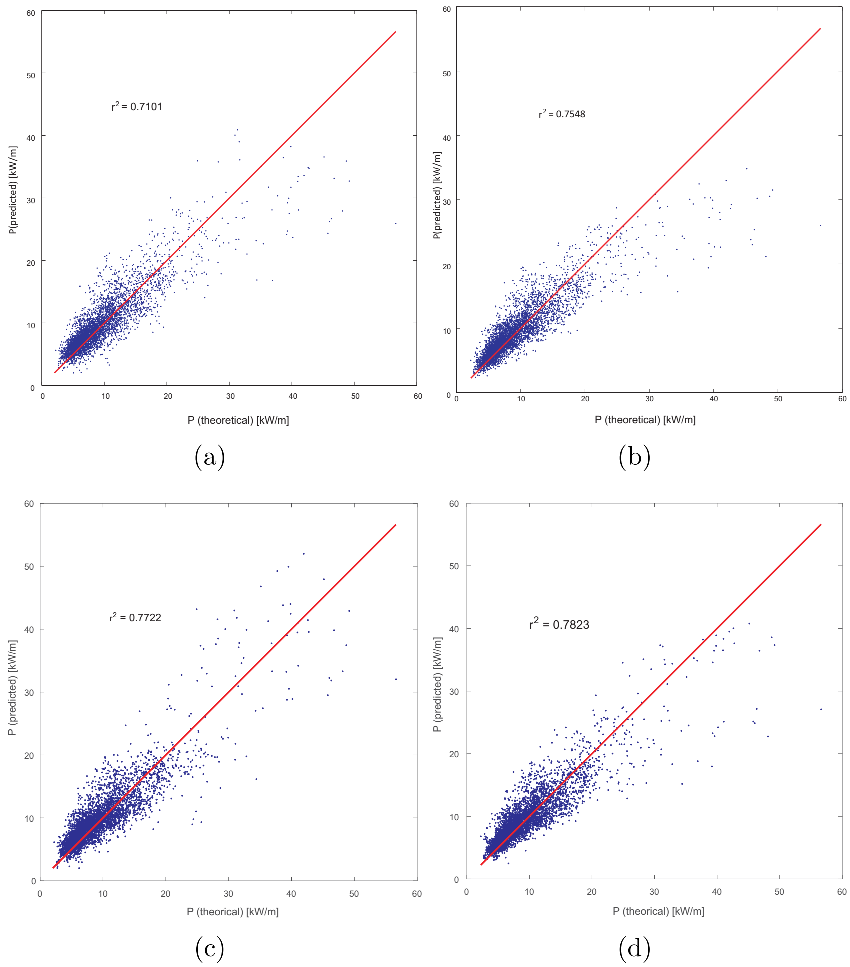

Figure 8: Scatter plots in the problem of P prediction in tackled by the ELM and SVR with feature selection by the GGA-ELM; (a) ELM; (b) SVR; (c) ELM with Bayesian optimization; (d) SVR with Bayesian optimization.

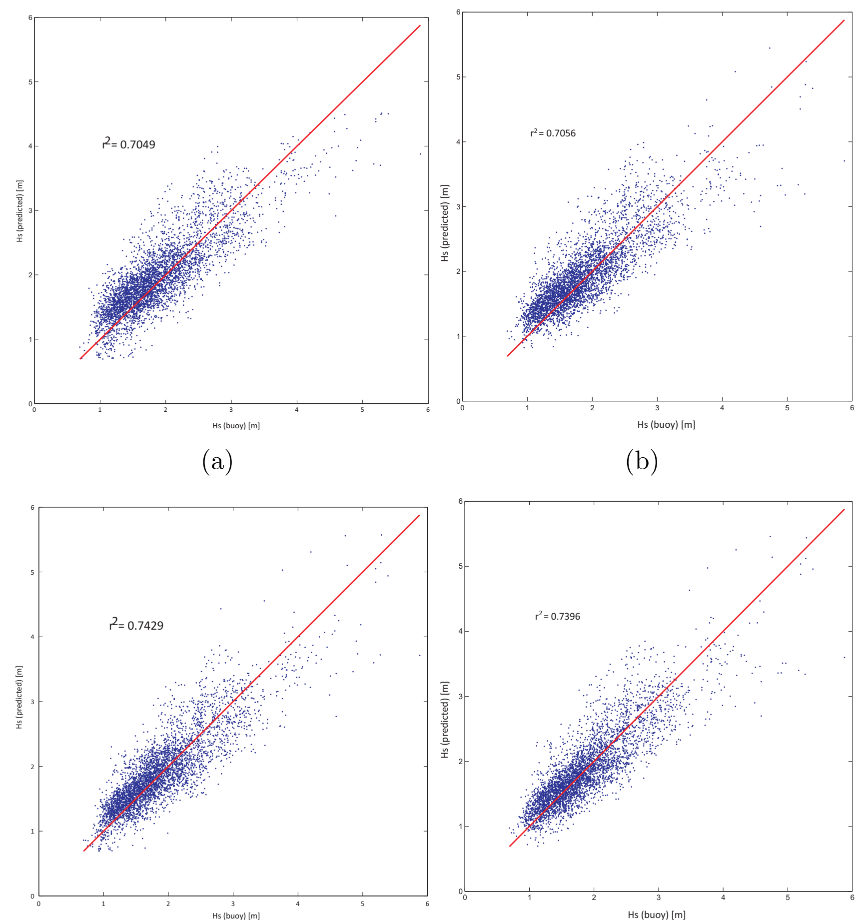

Figure 9: Scatter plots in the problem of H_s prediction in tackled by the ELM and SVR with feature selection by the GGA-ELM; (a) ELM; (b) SVR; (c) ELM with Bayesian optimization; (d) SVR with Bayesian optimization.