Genetic algorithm based support vector machine regression in predicting wave transmission of horizontally interlaced multi-layer moored floating pipe breakwater

S.G. Patil\textsuperscript{a,}\textsuperscript{*}, S. Mandal\textsuperscript{b}, and A V Hegde\textsuperscript{c}

\textsuperscript{a} Department of Built and Natural Environment, Caledonian College of Engineering, PO Box: 2322, CPO Seeb, PC 111, Sultanate of Oman.
\textsuperscript{b} Ocean Engineering Division, National Institute of Oceanography, Goa, 403 004, India.
\textsuperscript{c} Department of Applied Mechanics and Hydraulics, National Institute of Technology Karnataka, Surathkal, Srinivasnagar, Mangalore 575025, India.

*Corresponding author.
E-mail addresses: sanras5@gmail.com (S.G. Patil), smandal.nio@org (S. Mandal), arkalvittal@gmail.com (A. V. Hegde).

Abstract: Planning and design of coastal protection works like floating pipe breakwater require information about the performance characteristics of the structure in reducing the wave energy. Several researchers have carried out analytical and numerical studies on floating breakwaters in the past but failed to give a simple mathematical model to predict the wave transmission through floating breakwaters by considering all the boundary conditions. Computational Intelligence techniques, such as, artificial neural networks (ANN), fuzzy logic, genetic programming and support vector machine (SVM) are successfully used to solve complex problems. In the present paper, a hybrid genetic algorithm tuned support vector machine regression (GA-SVMR) model is developed to predict wave transmission of horizontally interlaced multilayer moored floating pipe breakwater (HIMMFPB). Furthermore, optimal SVM and kernel parameters of GA-SVMR models are determined by genetic algorithm. The GA-SVMR model is trained on the data set obtained from experimental wave transmission of HIMMFPB using regular wave flume at Marine Structure Laboratory, National Institute of Technology, Karnataka, Surathkal, Mangalore, India. The results are compared with ANN and Adaptive Neuro-Fuzzy Inference System (ANFIS) models in terms of correlation coefficient, root mean square error and scatter index. Performance of GA-SVMR is found to be reliably superior. b-spline kernel function performs better than other kernel functions for the given set of data.

Keywords: Support vector machine, Genetic algorithm, artificial neural network, ANFIS, Floating breakwater, HIMMFPB, Wave transmission.
1. Introduction

Floating breakwaters are well accepted in recent years because of their basic advantages, such as, flexibility, easy mobilization, installation, and retrieval. The system can be fabricated in land, towed to the site, and installed along any desired alignment with ease. In addition, they have several desirable characteristics, such as, comparatively small capital cost, adoption to varying harbour shapes and sizes, short construction time and freedom from silting and scouring. Floating breakwaters could also be utilized to meet location changes, extent of protection required or seasonal demand. They can be used as a temporary protection for offshore activities in hostile environment during construction, drilling works, salvage operation, etc. Hence, it is necessary to study a detailed investigation of proposed floating breakwater.

Several researchers in the past have carried out experimental and numerical investigations on different types of floating breakwaters, such as, horizontal, sloping, A-type, Y-type, Cage, pontoon, tires, pipes etc. [1-11], However, there is a lack of a simple mathematical model to predict breakwater performance characteristics, such as the transmission coefficient, which is defined as the ratio of the transmitted wave height past the breakwater to the incident wave height on the breakwater. It is also found that most of the numerical methods have been attempted on simple box-type rectangular floating breakwaters or spar buoy floating breakwaters. These studies are carried out considering a floating breakwater in basic form with some assumptions common in hydrodynamics, which shows less improvement. Till now, there has not been available a simple mathematical model to predict a wave transmission through floating breakwaters by considering all the boundary conditions. Also, for floating pipe breakwaters the energy dissipation process depends on various other factors like pipe interference effect, the spacing between the pipes and number of layers. As the effect of all these factors on transmission and forces in the moorings is not clearly understood, it will be extremely difficult to quantify them mathematically. Still it is a complex problem.

Computational intelligence techniques, such as, Artificial Neural Networks (ANN), Fuzzy logic (FL), Genetic Programming (GP), Support Vector Machines (SVM) or combinations of these techniques are successfully used to solve complex problems associated with coastal/ocean engineering. Among these techniques, ANN is widely used in coastal/ocean engineering to predict ocean wave parameters like wave height, wave period, impact wave force etc. [12-15]. The most significant features of neural
networks are the extreme flexibility due to learning ability and the capability of non-linear function approximations. This has made ANN very popular in recent years, and further this technique has provided promising results in prediction of tidal levels [16], damages to coastal structures [17], depth of eroded caves in a seawall [18], seabed liquefaction [19], storm surges [20], etc. According to Shahidi and Mahjoobi [21] ANNs are not as transparent as semi-empirical regression based models. In addition, neural network approach needs to find network parameters, such as number of hidden layers and neurons by trial and error, which is time consuming. To overcome the problems inherent in ANN training procedures Jeng et al., [19] adopted the concept of genetic algorithm based training of ANN models, which provided accurate results for determining maximum liquefaction depth in a real world application. It is also noticed that apart from improving the performance of ANN, computational effort and time needed for training and testing the model is significantly reduced compared to traditional methods [17].

When the performance of ANN alone is poor in mapping input-output relation, many researchers developed hybrid models by combining ANN with fuzzy system, ANN with numerical wave modeling [22, 23, 24], adaptive neuro-fuzzy inference system by Sylaios [23] for wind wave modeling, model trees by Shahidi and Mahjoobi [21] for prediction of significant wave height, etc., for example. ANN is a low level computational structure that performs well when dealing with raw data. The pure feed-forward back propagation learning process could easily be trapped into the local minima.

Recently fuzzy inference systems have become popular in solving complex engineering problems and are widely used in coastal/ocean engineering. However, fuzzy systems lack the ability to learn and cannot adjust themselves. Inspired by the idea of basing the fuzzy logic inference procedure on a feed forward network structure, Jang [25] proposed a fuzzy neural network model - the adaptive neuro-fuzzy inference system (ANFIS), which is a five-layer feed-forward neural network, which includes fuzzification layer, rule layer, normalization layer, defuzzification layer and a single summation neuron. It is a hybrid neuro-fuzzy technique that brings learning capabilities of neural networks to fuzzy inference system. An ANFIS uses a hybrid learning algorithm that combines the least-squares method and gradient descent principle [25, 26]. This hybrid model attracted many researchers to solve complex problems associated with coastal/ocean engineering [22, 27-29]. Ozger and Zekai [30] have adopted dynamic fuzzy approach to identify the effect of wind speed on wave characteristics variations in ocean wave generating system. Bakhtyar et al., [31] have concluded that the ANFIS model is more flexible
than the FIS model, with more options for incorporating the fuzzy nature of the real world system. Sylaios et al., [23] have used Takagi-Sugeno [32] rule based fuzzy inference system for forecasting wave parameters based on the wind speed, direction and the lagged wave characteristics. They used subtractive clustering method to identify the initial and final antecedent fuzzy membership functions. Yagci et al., [33] have used fuzzy logic method in breakwater damage ratio estimation. Erdik [34] has applied fuzzy logic approach in design of conventional rubble mound structures. Apart from above computational intelligence techniques, many authors have used various new approaches to solve complex coastal engineering problems like genetic programming by Gaur and Deo [35] for real time wave forecasting, Guven et al.,[36] for prediction of circular pile scour.

SVMs are the recently developed learning techniques that have gained enormous popularity in the field of classification, pattern recognition and regression. SVM works on structural risk minimization principle that has greater generalization ability and is superior to the empirical risk minimization principle as adopted in conventional neural network models. Han et al., [37] applied SVM for flood forecasting, Radhika and Shashi [38] for prediction of atmospheric temperature, Msiza, et al., [39] used ANN and SVR for water demand prediction, Rajasekaran et al.,[40] developed a support vector machine regression (SVMR) model for forecasting storm surges. They compared these results with numerical methods and ANN, which indicated that storm surges and surge deviations are efficiently predicted using SVMR. According to Mahjoobi and Mosabbeb [41], SVM creates a more reliable model with better generalization error, in comparison to ANN, they also reveal that SVMs do not over-fit, while ANNs may face such problem and need to deal with it.

Soft computing tools are used for other applications as discussed above. However, it is observed that there are hardly any applications of SVMs on wave transmission of floating breakwater. This fact leads us to use SVM models in this work. In the present paper, the performance of GA-SVMR models for predicting wave transmission coefficient of HIMMFPB is investigated. GAs are used to optimize the SVMR and kernel parameters. Results of GA-SVMR models are compared with that of ANN and ANFIS models.

The paper is organized as follows: section 1 starts with literature associated with floating breakwaters and applications of soft computing techniques in coastal engineering. Section 2 details wave transmission of floating breakwater and experimental HIMMFPB. Fundamentals of SVMR, GAs for
parameter selection and proposed GA-SVMR are detailed in section 3. Results and discussion are described in section 4. Conclusions and acknowledgments are presented in section 5 and 6, respectively.

2. Wave Transmission of Floating Breakwater

The design of floating breakwater is based on the principle that the wave energy is concentrated at the surface in deep water and the same energy is concentrated at below the surface in shallow water, which is to be dissipated. Therefore different types of floating breakwaters like Box, Pontoon, Mat, Tethered float and Pipe are becoming popular. The basic concept by which floating breakwater reduces wave energy include reflection, dissipation, interference and conversion of the energy into non-oscillatory motion. The prime factor in the construction of floating breakwater is to make the width of the breakwater in the direction of wave propagation greater than one-half the wavelength and preferably as wide as incident wavelength. Otherwise, the breakwater rides over the top of the wave without attenuating it. Pontoon and Box types of floating breakwaters belong to the category in which the wave attenuation is achieved by reflecting the wave energy. Mat and Tethered belong to the other category, in which wave energy dissipation is mainly due to drag from the resultant float in motion. Pipe breakwaters mainly dissipate the wave energy, and also partly reflect and transmit the waves. For effective reflection, the breakwater should remain relatively motionless and penetrate to a depth sufficient to prohibit appreciable wave energy from passing underneath.

2.1 Experimental HIMMFPB Model Setup and Data used

The development of floating breakwaters by various investigations has been influenced by certain important features; large masses, large moment of inertia, and the combinations of two. Most of the literature indicates that the parameter “relative width” influences greatly the wave attenuation characteristics of the breakwater. The details of the HIMMFPB are shown in Fig. (1) [42 - 45]. The breakwater comprises of the rigid Poly Vinyl Chloride (PVC) pipes. These pipes are placed parallel to each other with certain spacing between them in each layer and the adjacent layers are oriented at right angles to each other, so as to form an interlacing. Longitudinal pipes are placed along the direction of propagation of waves and transverse pipes are placed and tied perpendicular to longitudinal pipes. The length of the longitudinal pipes defines the width of the breakwater. It is felt that with proper number of layers, spacing of pipes and relative breakwater width, it is possible to achieve a considerable and
effective attenuation of waves. Fig. (1) shows a pictorial representation of the HIMMFPB model in plan and section.

The wave-specific parameters and structure-specific parameters considered in the experiments are shown in Table 1. The experimental study carried out by Kamat [45] shows hydrodynamic characteristics of horizontally interlaced three and five layer floating breakwater systems, in which wave transmission is less for five layer systems. These experimental data are divided into two sets, one for training and other for testing the GA-SVMR models (Table 2). The input parameters that influence the wave transmission \(K_i\) of floating breakwater, such as, spacing of pipes relative to pipe diameter \((S/D)\), breakwater width relative to wave length \((W/L)\), incident wave relative to water depth \((H_i/d)\), incident wave relative to wave length \((H_i/L)\) are used to train GA-SVMR models as shown in Table 2.

Experimental data on \(W/L\), \(H_i/d\), \((S/D)\), \(H_i/L\) and \(K_i\) are shown in Fig. (2). From this experimental data, first 2131 data points are used to train GA-SVMR models and the remaining 813 data points are used for testing the models.

3. Hybrid Genetic Algorithm Tuned Support Vector Machine Regression (GA-SVMR)

3.1 Fundamentals of Support Vector Machine Regression (SVMR)

Vapnik [46] proposed the support vector machines (SVMs), which is based on statistical learning theory. The basic idea of support vector machines is to map the original data \(x\) into a feature space with high dimensionality through a non-linear mapping function and construct an optimal hyper-plane in new space. Hence, given a set of data \(S=\{(x_i,d_i)\}_{i=1}^{N}\), where \(x_i\) is the input data set, \(d_i\) is the desired result, and \(N\) corresponds to the size of the data set. Then, according to Smola and Scholkopf [47], the SVM regression function is expressed as

\[
y = f(x) = w_i \phi_i(x) + b
\]

Where \(\phi_i(x)\) is the non-linear function in feature of input \(x\), and both \(w_i\) and \(b\) are coefficients, which are estimated by minimizing the regularized risk function as expressed below:

\[
\text{Minimize: } R(C) = \frac{1}{2} \|w\|^2 + C \frac{1}{N} \sum_{i=1}^{N} L_e(d_i, y_i)
\]
where;

\[
L_\varepsilon (d_i, y_i) = \begin{cases} 
|d_i - y_i| - \varepsilon, & |d_i - y_i| \geq \varepsilon, \\
0, & \text{others} 
\end{cases}
\]

(3)

The first term in equation 2 is called regularized term, measures the flatness of the function. The second term is the empirical error measured by the \( \varepsilon \)-insensitive loss function, which is defined as equation 3. \( C \) and \( \varepsilon \) are user determined parameters, \( d_i \) is the actual value at period \( i \), \( y_i \) is the forecasted value at period \( i \), and \( C \) is a weighing parameter considered to specify the trade-off between the empirical risk and model flatness. Equation 3 defines a range where the loss will be zero if the forecasted value is within the \( \varepsilon \)-tube (Equation 3 and Fig. (3)). However, if the value is out of the \( \varepsilon \)-tube then the loss is the absolute value, which is the difference between forecasted value and \( \varepsilon \). Introducing two positive slack variables \( \zeta_i \) and \( \zeta_i^* \) in Equation 3, it is possible to transform it into a primal objective function given by:

\[
\text{Minimize: } R(w, \zeta_i, \zeta_i^*) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} (\zeta_i + \zeta_i^*)
\]

(4)

subject to the constraints;

\[
\begin{align*}
& d_i - w_i \phi(x_i) - b \leq \varepsilon + \zeta_i, \\
& w_i \phi(x_i) + b - d_i \leq \varepsilon + \zeta_i^*, \\
& \text{where, } \zeta_i, \zeta_i^* \geq 0, \quad i = 1, 2, \ldots, N.
\end{align*}
\]

\( \zeta_i \) and \( \zeta_i^* \) represents the distance from the actual values to the corresponding boundary values of \( \varepsilon \)-tube.

The key idea is to construct the Lagrange function from the primal objective function and corresponding constraints by introducing the dual set of variables,

\[
L(w, \zeta_i, \zeta_i^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} (\zeta_i + \zeta_i^*) - \sum_{i=1}^{N} \alpha_i [w_i \phi(x_i) + b - d_i + \varepsilon + \zeta_i] \\
- \sum_{i=1}^{N} \alpha_i^* [d_i - w_i \phi(x_i) - b + \varepsilon + \zeta_i^*] - \sum_{i=1}^{N} (\beta_i \zeta_i + \beta_i^* \zeta_i^*)
\]

(5)

Equation 5 is minimized with respect to primal variables \( w_i, b, \zeta, \zeta^* \), and maximized with respect to non-negative Lagrangian multipliers \( \alpha_i, \alpha_i^*, \beta_i, \beta_i^* \). Finally, Karush-Kuhn-Tucker conditions are applied to the regression, and Equation 5 thus yields the dual Lagrangian.
\[ J(\alpha_i, \alpha_i^*) = \sum_{i=1}^{N} d_i (\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) K(x_i, x_j) \]  

(6)

subject to the constraints;

\[ \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i \alpha_i^* \leq C, \quad i = 1, 2, ..., N. \]

In Equation 6, \( \alpha_i \) and \( \alpha_i^* \) are called Lagrangian multipliers that satisfy equalities, \( \alpha_i \times \alpha_i^* = 0 \). After calculating \( \alpha_i \) and \( \alpha_i^* \), an optimal desired weights vector of the regression hyper-plane is represented as

\[ w^* = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x_i, x_j) \]  

(7)

Therefore, the regression function is expressed as,

\[ f(x, \alpha, \alpha^*) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x_i, x_j) + b \]  

(8)

Here, \( K(x_i, x_j) \) is called the kernel function. The value of the kernel equals the inner product of two vectors \( x_i \) and \( x_j \) in the feature space \( \phi(x_i) \) and \( \phi(x_j) \), i.e., \( K(x_i, x_j) = \phi(x_i) \times \phi(x_j) \). The role of the kernel function simplifies the learning process by changing the representation of the data in the input space to a linear representation in a higher-dimensional space called a feature space. A suitable choice of kernel allows the data to become separable in the feature space despite being non-separable in the original input space. This allows us to obtain non-linear algorithm from algorithms previously restricted in handling linearly separable datasets. The function that satisfies Mercer’s condition by Vapnik [48] can be used as the kernel function.

In the present paper, we have experimented with the six kernels as shown in Table 3. The linear kernel function is used for linear SVM model, whereas the polynomial, radial basis function (rbf), exponential radial basis function (erbf), spline and b-spline kernels are used for non-linear SVM models. According to Karatzoglou and Meyer [49], Gaussian radial basis function kernel is the general purpose kernel used when there is no prior knowledge about the data. The linear kernel is useful, when dealing with large sparse data vectors, as usually the case in text categorization. The polynomial kernel is popular in image processing, whereas, the spline kernels typically perform well in regression. Selection
of two kernel parameters \((d, \gamma)\) and support vector machine parameters \((C, \varepsilon)\) of a SVM model is significant in accuracy of the forecasting, where, the parameter \(d\) represents the degree of polynomial and b-spline kernel functions, whereas, \(\gamma\) is the width of rbf and erbf kernel functions. The generalization performance of GA-SVMR depends on a good setting of \(C, \varepsilon\) and kernel parameters \(d\) and \(\gamma\). Parameter \(C\) determines the trade-off between the model complexity (flatness) and the degree to which deviations larger than \(\varepsilon\) tube [50, 51]. If \(C\) is too large (infinity), then the objective is to minimize the empirical risk only, without regard to the model complexity [52]. In the present study, quadratic loss function is used. The main idea of using this loss function is to ignore the errors, which are situated within the certain distance of the true value. Parameter \(\varepsilon\) controls the width of the \(\varepsilon\)-insensitive zone, which is used to fit the training data. The number of support vectors \((\text{nsv})\) used to construct regression function depends on \(\varepsilon\), the big \(\varepsilon\), the fewer support vectors are selected and results in data compression [53]. To optimize these parameters for better generalization of SVM model, SVM model is hybridized with GAs. Section 3.2 details the genetic algorithm in parameter selection, whereas section 3.3 details interface of genetic algorithm with support vector machine regression to obtain the best GA-SVMR model.

3.2 Genetic algorithm for selecting parameters in the SVMR Model

Genetic algorithms are search methods based on principles of natural selection and genetics [54]. The algorithm is based on the principle of the survival of the fittest, which tries to retain genetic information from generation to generation. In this paper, GAs is used to search for better combination of \(C, \varepsilon\) and kernel parameters \((d\) and \(\gamma\)) to maximize the generalization performance of SVMR model. The procedure of genetic algorithms in parameter selection is shown in Fig. (4), whereas Fig. (5) shows the proposed GA-SVMR model. The codes are written in MATLAB 7 Release 14. The steps involved in GA for selecting SVMs and kernel parameters are as follows:

**Step 1.** (Initialization): In the present paper initial population of chromosomes is generated randomly. Population size is set to 50. The chromosomes are real coded string, consist of SVMs parameters \(C, \varepsilon\) and kernel parameters \((d\) and \(\gamma\)).

**Step 2.** (Evaluating fitness): In this step fitness of each chromosome is evaluated. In the present paper, a
negative normalized mean square error is used as the fitness function, which is defined as:

\[
\text{Fitness Function} = \frac{1}{\sigma^2 N} \sum_{i=1}^{N} (d_i - y_i)^2
\]  

(9)

Where, \( \sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (d_i - \bar{d}_i)^2 \)

\(N\) is the total number of data in the test set, \( \bar{d}_i \) is the mean of the actual value, \( d_i \) is the actual value, and \( y_i \) is the predicted value.

**Step 3. (New population):** In this step new population is created by repeating following steps until the new population is complete

*i) [Selection]:* In the present study, two parent chromosomes from a population are selected according to fitness function (Equation 9). The roulette wheel selection principle [54] is used to select chromosomes for reproduction

*ii) [Crossover]:* Here with crossover probability crossover of the parents is done to form new offspring’s (children). In cross over, chromosomes are paired randomly. The intermediate crossover principle is used and offspring’s are produced according to the following rule

\[
\text{Offspring} = \text{parent}_1 \pm \alpha (\text{parent}_2 - \text{parent}_1)
\]  

(10)

Where \( \alpha \) is the scaling factor chosen uniformly at random over an interval \([-d, 1+d]\). In the present study \(d\) is chosen as 0.25.

*iii) [Mutation]:* After cross over operation is performed the string is subjected to mutation operation, this is to prevent falling all solutions of the population into local optimum of solved problem. The variable in the string to be mutated is selected randomly, where incremental operator is used. The rate of crossover and mutation is determined by probabilities. In the present paper, the probabilities of crossover and mutation are set to 0.8 and 0.05 respectively.

*iv) [Accepting]:* Accept and place new offspring in the new population.
Step 4. (Replace): Here new generated population is used for a further run of the algorithm.

Step 5. (Stop condition): If the end condition is satisfied, stop, and return the best solution in current population. Otherwise,

Step 6. (Loop): Go to step 2.

3.3 The proposed GA-SVMR Model

In the present study, MATLAB support vector machine toolbox [50] is interfaced with genetic algorithm to optimize the SVMs and kernel parameters simultaneously for better generalization of the proposed GA-SVMR model. Six GA-SVMR models were developed by using six kernel functions (Table 3). In order to study, the performance of each kernel in predicting wave transmission of HIMMFPB, GA-SVMR is trained by applying these kernel functions. For training, experimental data set is used (Fig. (2)) and is divided into two groups one for training and other for testing (Table 2). Fig. (5) illustrates the proposed GA-SVMR model. In the first stage training input, training target, kernel function, and range of kernel and SVM parameters are fed to the system. GA generates the initial population that would be used to find optimum factors of kernel functions and SVMs. In the second stage, the system performs typical SVM process using assigned value of the factors in the chromosomes, and calculates the performance of each chromosome. The performance of each chromosome is calculated using fitness function for GAs given in Equation 9.

In the present study, the main goal is to find the best parameters that produce the most accurate prediction. If the calculated fitness value satisfies the terminal condition in GAs, then the optimal parameters are selected, otherwise, the new generation of the population is produced by applying genetic operators, such as, selection, crossover, and mutation. After the production of new generation, the training process with calculation of the fitness value is performed again. From this point, stage two and stage three are iterated again and again until the stopping conditions are satisfied. Once the stopping condition is satisfied, the genetic search finishes and the chromosomes that shows the best performance in the last population is selected as the final result. These optimize SVMs and kernel parameters are shown in Table 4. In the forth and final stage optimized parameters obtained by GA are tested with the test data. The final decision about the optimum models is not based on the training data, but on the testing data, as illustrated in Fig. (5). Once the testing is over, the six models with linear,
polynomial, rbf, erbf, spline and b-spline kernels are compared based on statistical measures to get the best model.

4. Results and Discussion

To study the effectiveness of the approach, statistical comparison of measured and predicted values of $K_i$, correlation coefficient $(CC)$ is used, which is defined as

$$CC = \frac{\sum_{i=1}^{N} (K_{imi} - \overline{K_m}) (K_{ypi} - \overline{K_p})}{\sqrt{\sum_{i=1}^{N} (K_{imi} - \overline{K_m})^2} \times \sqrt{\sum_{i=1}^{N} (K_{ypi} - \overline{K_p})^2}}$$

(11)

Where $K_{imi}$ and $K_{ypi}$ represents the measured and predicted wave transmission coefficient, respectively, $\overline{K_m}$ and $\overline{K_p}$ are the mean value of measured and predicted observations, $N$ is the number of observations. Higher the CC value better is the agreement between the measured and predicted values of $K_i$. Apart from this, other statistical measures computed are root-mean-square error $(RMSE)$, and scatter index $(SI)$. These are defined as

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (K_{imi} - K_{ypi})^2}$$

(12)

$$SI = \frac{RMSE}{\overline{K_m}}$$

(13)

Statistical measures computed using train and test data for GA-SVMR are shown in Table 5. Train and test data are used to compare the models results. The trained and test results (CCs) of six models are shown in Table 5 and Fig. (6-11).

In comparison to all models, linear kernel function shows poor generalization performance (CC Train = 0.8964 and CC Test = 0.8924) in prediction of $K_i$ for HIMMFPB with $SI$, 0.12375 and 0.12925 for train and test data respectively. Number of support vectors used in GA-SVMR with linear kernel function is 2131, which is 100%. Similarly, GA-SVMR with non-linear kernel functions also used 100% of support vectors, which indicates that every training input is utilized as support vector. This clearly proves that, there is no noise in the training data set, but there is non-linearity and
complexity associated in mapping input and output parameters of HIMMFPB. Increasing the $C$ will disturb the solution, but it can be helpful for other kernels like spline kernel, where $C$ is 40000 as shown in Table 4. In case of b-spline kernel $C$ is 15, whereas for erbf it is 6. For polynomial and linear kernel it is 96 and 100 respectively. A b-spline kernel function has better generalization performance with $RMSE$ 0.02286 and 0.03629 for train and test data respectively. Whereas, similar trend is shown by spline kernel function with slightly higher $RMSE$ 0.02896 and 0.03671 for train and test data respectively.

Correlation coefficient of GA-SVMR(b-spline) model (CC Train = 0.9897 and CC Test = 0.9741) is slightly better than GA-SVMR(spline) model, but considerably better than GA-SVMR(linear) model, whereas the performance of GA-SVMR(polynomial) model is better than GA-SVMR(rbf) and GA-SVMR(erbf) models with $SI$ 0.08824 for test data, whereas, for rbf and erbf kernels it is 0.09133 and 0.09655 respectively. In comparison to GA-SVMR model with b-spline and spline kernel functions, $SI$ and $RMSE$ is very high for GA-SVMR models with linear and erbf kernel function for test data (Table 5). It is noticed that the performance of these models depends on the better selection of SVM and kernel parameters. In case of polynomial kernel, the degree of the function $d$, when low; the function estimation is very bad, however, for higher $d$, performance is good. The optimal value of $d$ in case of polynomial kernel function obtained by GAs is 6 and for b-spline kernel function it is 2. The optimal width ($\gamma$) obtained by GAs in case of rbf and erbf kernel functions are 0.3 and 1 respectively. Kernel and SVM parameters obtained by GA (Table 4) is tested by using test data sets (Table 2), which show better generalization performance with highest CC Test = 0.9741 for GA-SVMR (b-spline) model.

The same data set has been used for estimating $K$, using ANN [55] and ANFIS [56]. CCs, $SI$ and $RMSE$ of $K$, are shown in Table 6. From Table 5 and 6, it is observed that the GA-SVMR (b-spline) model yields higher CCs as compared to that of ANN and ANFIS models, whereas $RMSE$ and $SI$ values are higher in ANN and ANFIS models as compared to GA-SVMR (b-spline) model. However the GA-SVMR model with linear kernel function has shown poor generalization, whereas, ANFIS model perform better than GA-SVMR models with polynomial, rbf and erbf kernel functions.

In SVM regression, the solution is unique for specific loss function, kernel type, and SVM and
kernel parameters. If we run the same program, the results will be exactly the same. The model is much more complex and cannot be used in other implementations, whereas the results will not be same in case of ANN and ANFIS models.

5. Conclusions

An application of hybrid genetic algorithm tuned support vector machine regression for prediction of wave transmission for HIMMFPB is presented in this paper. Our proposed model optimizes SVMs and kernel parameters simultaneously. The performance of GA-SVMR models is compared with ANN and ANFIS models. The results obtained shows that GA-SVMR with b-spline kernel functions performs better than ANN and ANFIS models.

The forecasting performance of GA-SVMR appears to be highly influenced by the choice of its kernel function, and the good setting of kernel and SVM parameters. The b-spline kernel function performed superior than other kernel.

It is also observed that parameter selection in the case of GA-SVMR has a significant effect on the performance of the model.

GA-SVMR can replace the ANN and ANFIS based models for wave transmission prediction of HIMMFPB.

GA-SVMR can be utilized to provide a fast and reliable solution in prediction of the wave transmission for HIMMFPB, thereby making GA-SVMR as an alternate approach to map the wave structure interactions of HIMMFPB.

6. Acknowledgements

The authors are grateful to the Director, and Head, Department of Applied Mechanics and Hydraulics, NITK, Surathkal, India for support and encouragement provided to them and for permission to publish the paper. Thanks are also due to Ministry of Earth Sciences, GOI for sponsoring the project on HIMMFPB at NITK, Surathkal, India.
References


List of Tables

Table 1 Range of wave-specific and structure-specific parameters used in HIMMFPB.

Table 2 Data used for training and testing the GA-SVMR models with input parameters.

Table 3 Kernel Functions.

Table 4 Optimal parameters for GA-SVMR models with different kernels.

Table 5 GA-SVMR models with statistical measures.

Table 6 ANN and ANFIS models statistical measures.
Table 1 Range of wave-specific and structure-specific parameters used in HIMMFPB

<table>
<thead>
<tr>
<th>Wave-specific parameters</th>
<th>Experimental range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incident wave height, $H_i$ (mm)</td>
<td>30, 60, 90, 120, 150, 180</td>
</tr>
<tr>
<td>Wave period, $T$ (s)</td>
<td>1.2, 1.4, 1.6, 1.8, 2.0, 2.2</td>
</tr>
<tr>
<td>Depth of water, $d$ (mm)</td>
<td>400, 450, 500</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Structure-specific parameters</th>
<th>Experimental range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diameter of pipes, $D$ (mm)</td>
<td>32</td>
</tr>
<tr>
<td>Ratio of spacing to diameter of pipes, $S/D$</td>
<td>2, 3, 4 and 5</td>
</tr>
<tr>
<td>Relative breakwater width, $W/L$</td>
<td>0.4-2.65</td>
</tr>
<tr>
<td>Number of layers, $n$</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 2 Data used for training and testing the GA-SVMR models with input parameters

<table>
<thead>
<tr>
<th>Model</th>
<th>Input Parameters</th>
<th>Number of data points for training</th>
<th>Number of data points for testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA-SVMR(linear)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GA-SVMR-polynomial</td>
<td>$(S/D), (W/L), (H_i/d), (H_i/L)$</td>
<td>2131</td>
<td>813</td>
</tr>
<tr>
<td>GA-SVMR(rbf)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GA-SVMR(ertf)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GA-SVMR(spline)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GA-SVMR(b-spline)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Table 3** Kernel Functions.

<table>
<thead>
<tr>
<th>Kernels</th>
<th>Functions</th>
<th>User defined kernel parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear</td>
<td>$K(x_i, x_j) = a_1x_ix_j + a_2$</td>
<td>For simplicity set $a_1 = 1$, $a_2 = 0$.</td>
</tr>
<tr>
<td>polynomial</td>
<td>$K(x_i, x_j) = ((x_i, x_j) + 1)^d$</td>
<td>$d$</td>
</tr>
<tr>
<td>rbf</td>
<td>$K(x_i, x_j) = \exp \left(-\frac{|x_i - x_j|^2}{2\gamma^2}\right)$</td>
<td>$\gamma$</td>
</tr>
<tr>
<td>erbf</td>
<td>$K(x_i, x_j) = \exp \left(-\frac{|x_i - x_j|^2}{2\gamma^2}\right)$</td>
<td>$\gamma$</td>
</tr>
<tr>
<td>spline</td>
<td>$K(x_i, x_j) = 1 + \left(\frac{x_i}{x_j}\right) + \frac{1}{2} \left(\frac{x_i}{x_j}\right)^3 - \frac{1}{6} \min\left(x_i, x_j\right)^3$</td>
<td>$-$</td>
</tr>
<tr>
<td>b-spline</td>
<td>$K(x_i, x_j) = B_{2d+1}(x_i - x_j)$</td>
<td>$d$</td>
</tr>
</tbody>
</table>

**Table 4** Optimal parameters for GA-SVMR models with different kernels.

<table>
<thead>
<tr>
<th>Kernel</th>
<th>$nsv$</th>
<th>$C$</th>
<th>$\epsilon$</th>
<th>$\gamma$</th>
<th>$d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear</td>
<td>2131</td>
<td>100</td>
<td>0.001</td>
<td>$-$</td>
<td>$-$</td>
</tr>
<tr>
<td>polynomial</td>
<td>2131</td>
<td>96</td>
<td>0.001</td>
<td>$-$</td>
<td>6</td>
</tr>
<tr>
<td>rbf</td>
<td>2131</td>
<td>150</td>
<td>0.001</td>
<td>0.3</td>
<td>$-$</td>
</tr>
<tr>
<td>erbf</td>
<td>2131</td>
<td>6</td>
<td>0.001</td>
<td>1</td>
<td>$-$</td>
</tr>
<tr>
<td>spline</td>
<td>2131</td>
<td>40000</td>
<td>0.05</td>
<td>$-$</td>
<td>$-$</td>
</tr>
<tr>
<td>b-spline</td>
<td>2131</td>
<td>15</td>
<td>0.05</td>
<td>$-$</td>
<td>2</td>
</tr>
</tbody>
</table>

**Table 5** GA-SVMR models with statistical measures.

<table>
<thead>
<tr>
<th>Model</th>
<th>CC Train</th>
<th>CC Test</th>
<th>Train Data</th>
<th>Test Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$RMSE$</td>
<td>$SI$</td>
</tr>
<tr>
<td>GA-SVMR (linear)</td>
<td>0.8964</td>
<td>0.8924</td>
<td>0.07072</td>
<td>0.12375</td>
</tr>
<tr>
<td>GA-SVMR (polynomial)</td>
<td>0.9568</td>
<td>0.9513</td>
<td>0.04638</td>
<td>0.08116</td>
</tr>
<tr>
<td>HGASVMR (rbf)</td>
<td>0.9563</td>
<td>0.9478</td>
<td>0.04662</td>
<td>0.08157</td>
</tr>
<tr>
<td>HGASVMR (erbf)</td>
<td>0.9640</td>
<td>0.9416</td>
<td>0.04253</td>
<td>0.07443</td>
</tr>
<tr>
<td>HGASVMR (spline)</td>
<td>0.9834</td>
<td>0.9735</td>
<td>0.02896</td>
<td>0.05068</td>
</tr>
<tr>
<td>GA-SVMR (b-spline)</td>
<td>0.9897</td>
<td>0.9741</td>
<td>0.02286</td>
<td>0.03900</td>
</tr>
</tbody>
</table>
Table 6 ANN and ANFIS models with statistical measures.

<table>
<thead>
<tr>
<th>Model</th>
<th>CC Train</th>
<th>CC Test</th>
<th>Train Data</th>
<th>Test Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>RMSE</td>
<td>SI</td>
</tr>
<tr>
<td>ANN [55]</td>
<td>0.9537</td>
<td>0.9488</td>
<td>0.05176</td>
<td>0.09058</td>
</tr>
<tr>
<td>ANFIS [56]</td>
<td>0.9723</td>
<td>0.9635</td>
<td>0.03727</td>
<td>0.06522</td>
</tr>
</tbody>
</table>
List of Figures

Fig. (1). HIMMFPB model setup.

Fig. (2). Experimental data on \( W/L, H_i/d, S/D, H_i/L \) and \( K_i \).

Fig. (3). The loss margin setting corresponds to one dimensional linear SV machine.

Fig. (4). GAs procedure.

Fig. (5). Flow chart of GA-SVMR.

Fig. (6). Comparison of predicted and measured \( K_i \) for GA-SVMR (linear) model.

Fig. (7). Comparison of predicted and measured \( K_i \) for GA-SVMR (polynomial) model.

Fig. (8). Comparison of predicted and measured \( K_i \) for GA-SVMR (rbf) model.

Fig. (9). Comparison of predicted and measured \( K_i \) for GA-SVMR (erbf) model.

Fig. (10). Comparison of predicted and measured \( K_i \) for GA-SVMR (spline) model.

Fig. (11). Comparison of predicted and measured \( K_i \) for GA-SVMR (b-spline) model.
Fig. (1). HIMMFPB model setup.
Fig. (2). Experimental data on $W/L$, $H_i/d$, $S/D$, $H_i/L$ and $K_r$. 
Fig. (3). The loss margin setting corresponds to one dimensional linear SV machine.

Fig. (4). GAs procedure
Fig. (5). Flow chart of GA-SVMR.
**Fig. (6).** Comparison of predicted and measured $K_t$ for GA-SVMR (linear) model.
Fig. (7). Comparison of predicted and measured $K_t$ for GA-SVMR (polynomial) model.
Fig. (8). Comparison of predicted and measured $K_t$ for GA-SVMR (rbf) model.
Fig. (9). Comparison of predicted and measured $K_t$ for GA-SVMR (erbf) model.
Fig. (10). Comparison of predicted and measured $K_t$ for GA-SVMR (spline) model.
Fig. (11). Comparison of predicted and measured $K_t$ for GA-SVMR (b-spline) model.

CC Train $= 0.9897$

CC Test $= 0.9741$