IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received May 25, 2020, accepted June 5, 2020, date of publication June 15, 2020, date of current version June 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3002563

A Genetic Programming-Driven Data Fitting Method

HAO CHEN[1,2], ZI YUAN GUO[1], HONG BAI DUAN[1], AND DUO BAN[1]

[1]School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an 710121, China
[2]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an 710121, China

Corresponding author: Hao Chen (chenhao@xupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61876138 and Grant 61203311, in part by the Natural Science Basic Research Program of Shaanxi Province of China under Grant 2019JM-365, in part by the Scientific Research Program Funded by Shaanxi Provincial Education Department of China under Grant 17JK0701, in part by the Science Foundation of the Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing under Grant XUPT-KLND(201804), and in part by the Innovation Funds of Xi'an University of Posts and Telecommunications under Grant CXJJ2017043 and Grant CXJJLI2018017.

ABSTRACT Data fitting is the process of constructing a curve, or a set of mathematical functions, that has the best fit to a series of data points. Different with constructing a fitting model from same type of function, such as the polynomial model, we notice that a hybrid fitting model with multiple types of function may have a better fitting result. Moreover, this also shows better interpretability. However, a perfect smooth hybrid fitting model depends on a reasonable combination of multiple functions and a set of effective parameters. That is a high-dimensional multi-objective optimization problem. This paper proposes a novel data fitting model construction approach. In this approach, the model is expressed by an improved tree coding expression and constructed through an evolution search process driven by the genetic programming. In order to verify the validity of generated hybrid fitting model, 6 prediction problems are chosen for experiment studies. The experimental results show that the proposed method is superior to 7 typical methods in terms of the prediction accuracy and interpretability.

INDEX TERMS Data fitting, hybrid model, genetic programming, tree coding, interpretability.

I. INTRODUCTION

The goal of constructing a data fitting model is to seek a set of functions, which can describe the approximate correlation among a group of variables, and subject to constraints. It can be acted as a kind of data characterization or prediction tool. Generally, this method can be broadly divided into two categories, the model with a concrete function expression and the model based on some intelligent calculation approaches. The polynomial model and neural network are the typical one of former and latter respectively. In recent years, the ensemble learning and deep learning have been applied to deal with data fitting problem and show outstanding performance. However, the training process of them is relatively complicated. More importantly, the training-driven model turns out to be a black box which cannot showcase the coupling relationship among variables, making it difficult to comprehend and further utilize. Accordingly, the model with concrete function expression still has its advantages. But, the traditional methods, such as polynomial model, require a prior hypothesis including the type and number of used functions. In most cases, these are unknowable. And, a group of optimized parameters are also desired. Therefore, how to generate a fitting model with reasonable structure while optimizing related parameters has become a key problem.

It is found that the hybrid fitting model with lower complexity and higher fitting accuracy can be constructed by mixing different types of functions. But, constructing such a model firstly calls for mechanisms with more effective coding expression and optimization ability. Concerning this issue, this paper proposes a method for constructing the hybrid fitting model based on representation by tree coding and co-optimization of model structure and parameters by evolutionary search. Major contributions of this paper include:

1) The improved expression tree coding mechanism is proposed to express hybrid fitting model. In this coding mechanism, each node is composed of structure part and multiplier factor part. When its structure changes, the variable

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wang.

length of tree coding makes it possible to express the new model flexibly, which lay the foundation for searching and optimizing the model. Moreover, compared with traditional coding, the complexity of improved expression tree coding is reduced.

2) The optimization mechanism of hybrid fitting model based on improved genetic programming (GP) is proposed. This mechanism for co-optimization of model structure and parameters by evolutionary search, which can improve fitting accuracy of the model, reduce its complexity and make it possible to enhance its interpretability.

The remaining part of this work is organized as follows. Analysis of related research on fitting model and its optimization mechanism in Section II. Introduction of the proposed method for tree coding expression and relevant evolution and optimization of hybrid fitting model in Section III. Then, the results and its discussion in Section IV. Finally, some conclusion and future work in Section V.

II. RELATED WORKS

This section focuses on the discussion of related works of the data fitting method based on intelligent computing, fitting approaches with explicit function expression, and the mechanism for optimizing them.

A. DATA FITTING MODEL BASED ON INTELLIGENT COMPUTING METHOD

With the development of machine learning, the data fitting model based on the intelligent computing method has been widely applied. The support vector machine (SVM), for instance, is a mature means. Karimi *et al.* [1] developed binary SVM model for urban expansion prediction by selecting the most appropriate kernel function and its parameters. Sousa *et al.* [2] used the genetic algorithm to optimize the SVM model that helps forecast the classification and recovery rate of urban waste. Although SVM can solve nonlinear and local minimum problems, it is difficult to deal with a ton of data [3]. Ensemble learning as a practical method, such as random forest (RF), eXtreme gradient boosting (XGBoost) have been developed on the basis of Bagging and Boosting [4]. In [5], RF was used to construct model for predicting mortality rate of patients suffering acute renal injury, and in [6], a new ultra-short-term offline prediction model of photovoltaic characteristics based on RF was proposed. The results show that the prediction of RF is highly accurate, but the final result is limited by the prediction performance of each decision tree. In [7], XGBoost, a typical boosting algorithm, is used to avoid overfitting problems and establish an efficient energy load prediction model in residential buildings. Based on XGBoost, a C-A-XGBoost sales prediction model was proposed for focusing on the characteristics of commodity sales and the trend fitting of data series [8]. The experimental results suggest that the prediction is more accurate. Neural network is an effective nonlinear data fitting method [9], [10]. In recent years, with the development of convolutional neural network (CNN), the data prediction model based on it has been more widely used. In [11], a CNN-based framework for predicting the next day's direction of movement for the indices of S&P 500, NASDAQ, DJI, NYSE, and RUSSELL.

A data fitting model combining linear regression and the deep belief network model has been proposed [12]. In addition, the long short-term memory network (LSTM) has special structure of memory and gate, which is also frequently used to solve problems of prediction [13]–[15].

The data fitting method based on the intelligent computing shows excellent performance, but its working mechanism is complex, especially the model generated by training, which is a black box and cannot describe the detailed relations between different variables in the data. In many tasks of data fitting and prediction, interpretability of the model is of great significance. For example, the reference [16] pointed out that the energy consumption model of conveyor based on BP neural network is not conducive to describe the problems of controlling optimization, while the model on the basis of function expression is more reasonable.

B. DATA FITTING MODEL BASED ON FUNCTION EXPRESSION

The data fitting model based on function expression, in addition to the simpler structure, can express the coupling relationship between different variables in a clearer way as well. The polynomial model, a variant of the linear model, is a typical example adaptable to the nonlinear relationship [17]. The Gaussian distribution model is widely applied for its robustness and computational efficiency [18]. In addition, the Lasso regression can effectively deal with problems of high-dimensional data by constructing penalty function to obtain a more detailed model [19], [20]. A prediction method for wind power combining Lasso regression [21] shortens computing time greatly. Combined with Lasso regression to predict power consumption in [22], the output of Lasso regression shows that the power consumption of Guangdong Province is closely related to the historical consumption, the proportion of the secondary industry and the permanent population. In [23], a linear piecewise fitting model is given to forecast yield automatically from temperature, reactor volume and reactant concentration. Due to the complexity of chemical reaction, it is difficult for experts to make clear of the rules in yield prediction, and as it pointed out, the piecewise fitting model is easier to understand than SVR. An effective algorithm was proposed in [24] to identify the key segmentation features and the number of final segmentation points. Each segment was fitted with a multivariate linear regression function. But when the continuous trial is taken to automatically determine the number of data areas, the calculation is inefficient and may lead to over fitting.

Before being built, the data fitting model based on function expression generally calls for given structural hypothesis of the model, and following identification of relevant parameters. Such model has a simple working mechanism and clear expression of practical problems. However, for an unknown problem, it is often difficult to offer a reasonable structural hypothesis, and the optimization of related parameters will also directly affect the performance of the model.

C. OPTIMIZATION MACHANISM FOR MODEL PARAMETERS

In [25], the thermal error compensation model of the machine tool based on the exponential model in use of the least square method to optimize the estimation equation of axial deformation of spindle and time. In [26], the quasi newton method was used to optimize the parameters of multivariate nonlinear regression model, and obtained the regression model of dry matter the potato contains. In [27], a multiple nonlinear regression model was established by studying the influence of various operation parameters on the thermal environment. By defining two objective functions, maximum exergy efficiency and the minimum total cost, then the downhill simplex method is used to optimize parameters. In recent years, the research of optimization regression model based on evolutionary algorithms has been developed rapidly. The regression equation between the stress of solder joint and the structural parameters was established [28], and the genetic algorithm (GA) optimizes the structural parameters of solder joint, the optimal combination of structural parameters with the minimum stress of solder joint is available. In [29], a prediction model of crude oil price based on wavelet transformation and multiple linear regression, and particle swarm optimization (PSO) is used to optimize the model parameters. *Chen et al.* [30] introduced particle calculation into PSO to optimize the nonlinear model composed of multiple regression models. *Sheng et al.* [31] adopted expectation maximization (EM) [32], a common approach to estimate of the optimal super parameters, to optimize the Gaussian mixture regression for estimating the charge of electric vehicles. Such studies at present mainly focus on optimizing parameters in the model, but research for a mixture of the better model structure and the related parameters has not been reported.

III. PROPOSED APPROACH

The interpretability of a model is critical for many data fitting and prediction tasks. Obviously, a model with clear function expression meets more of this requirement. This study found that the hybrid data fitting model (referred to as the hybrid model) mixed by different types of functions can make the model much less complicated while ensuring the fitting accuracy, which does more favor to comprehension and analysis of the data fitting model. But the main problem is that how to optimize the structure and parameters of the hybrid model.

A. ANALYSIS OF THE HYBRID MODEL

Suppose the data set S consists of N data points, which can be expressed as $S = \{X_i, Y_i\}$, $X_i = [x_1, x_2, \ldots, x_d]$, $i = 1, 2, \ldots, N$. In S, X_i is the input of the ith sample point and Y_i is the output, d stands for the dimension of X_i, and N *is* the number of samples.

Definition 1: The definition of the hybrid model ψ is as shown in Equation 1

$$\psi = \sum_{k=1}^{K} w_k g_k, \tag{1}$$

where K is the number of subfunctions, w_k is the multiplier factor of the k-th subfunction, g_k is the k-th subfunction composed of an exponential function, a logarithmic function, a Gaussian function and a power function.

Definition 2: Model error. The mean absolute error is used to evaluate the error of ψ, and the model error f_{error} can be expressed as

$$f_{error} = \frac{1}{n} \sum_{i=1}^{n} \left| Y_i' - Y_i \right|, \tag{2}$$

where Y_i is the actual value of the ith sample point in ξ, Y_i' is the calculated value about ψ in ξ, and n is the number of sample points in ξ, ξ is the observation sample set, $\xi \in S$.

Definition 3: Model complexity. The number of subfunctions K is the complexity of ψ.

In general, ψ is the superposition of multiple subfunctions, so more of them contributes to a more complex model.

TABLE 1. Comparisons of model complexity and fitting degree.

Fitting models	f_{error}	f_{node}	R^2
2-order polynomial model	6.219	**2**	0.151
6-order polynomial model	5.764	6	0.365
8-order polynomial model	3.331	8	0.763
12-order polynomial model	2.241	12	0.884
14-order polynomial model	2.251	14	0.882
5 Gaussian models	2.293	5	0.879
8-order polynomial + 2 exponential models	2.141	10	0.896
2-order polynomial + 2 Gaussian models	**0.963**	4	**0.998**

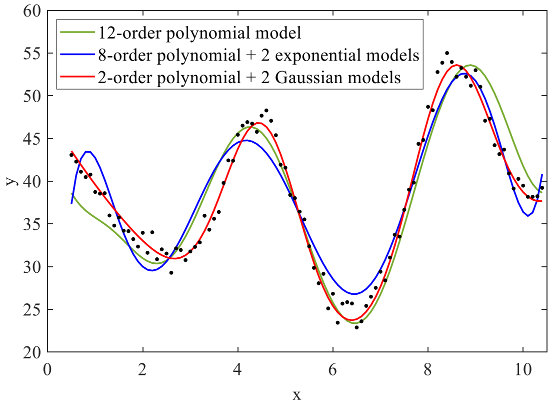

FIGURE 1. Comparison of fitting effects among different models.

Take the following example to illustrate the differences in different models. Table 1 shows the structure, error, complexity and the fitting degree of eight models that all adopt the least square method to optimize the relevant parameters. The fitting degree is calculated by decision coefficient (R^2). The larger it is, the better the fitting effect is. Figure 1 shows the fitting curves of three models with better performance.

From Table1, three findings stand out. Firstly, when the polynomial model is used to fit data, the fitting degree can be improved by moderately increasing complexity of the model. But when the latter reaches a certain level, the former will not be significantly enhanced, such as the change between the model of 12-order polynomial and the 14-order polynomial. Then, the Gaussian function leads to a higher fitting degree. For example, the fitting degree of the fitting model with five Gaussian subfunctions is close to that of the 12-order polynomial model. Finally, the hybrid model helps to improve the fitting degree and reduce complexity of the model by optimizing the combination structure of subfunctions. The last two models in Table 1 highlight this. In addition, the overall characteristics of the hybrid model manifest the effect of subfunction superposition. The model with a single type of function, when reflecting changes of data details, will improve overall performance inevitably at the cost of an increasingly complex model, such as the polynomial model. Instead, the hybrid model shown in Figure 1 works better featuring multiple subfunctions and less model complexity, which are proven to be very useful.

B. CODING MECHANISM OF HYBRID MODEL

The coding expression is the basis of constructing the hybrid model. To accommodate the search operation, the expression tree is used to encode the hybrid model. Besides, the decoding operation is a high-frequency calculation in the search process, complicated models will make calculation too large. In order to avoid it, the hybrid model can be encoded by the improved expression tree (I-ET).

In the I-ET coding mechanism, a node consists of two parts: the multiplier factor and the structure. In other words, the structure part (*sp*) of each node will be associated with a randomly generated multiplier factor part(*mp*). *sp* is the element selected from different type of sets made up of the function set $F = \{F_1, F_2, \ldots, F_m\}$ and terminal set $T = \{x_1, x_2, \ldots, x_d, c\}$. T is the set of input variables and the constant c. In fact, g_k in the hybrid model is the subtree composed of several nodes including *mp* and *sp*. As a special case, the Gaussian function constituting g_k is changing from the variable node to the Gaussian node with the certain probability P_{gs}. Figure 2 shows the coding structure between traditional expression tree coding and I-ET coding of node.

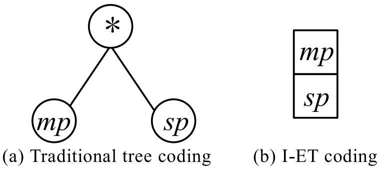

FIGURE 2. Coding structure of node.

Suppose the hybrid model ψ contains K subfunctions of which internal structure is not taken into consideration, $K - 1$ nodes are required for connection. In use of traditional expression tree coding and I-ET coding, the number of nodes required for a subfunction are 3 and 1 respectively. Therefore, for K subfunctions, the number of nodes is $4K - 1$ in the former coding, and $2K - 1$ in the latter. Undoubtedly, I-ET coding is less complicated.

Additionally, the flexibility of such coding helps to express structural changes in the model. As is shown in Figure 3, a new model can be obtained when the sub-function $\psi_z = 0.55N(x,8.5,0.9)$ is added to the original hybrid model $\psi_r = 7.5(0.01x^2\text{-}0.12\text{x}+0.51)+0.45N(x,4.5,\ 0.8)+0.96$. Obviously, an increase or decrease in the number of subfunctions in the original model only calls for adding or deleting the related branches of tree coding, without great change of overall coding structure.

C. OPTIMIZATION MECHANISM OF HYBRID MODEL

Genetic programming [33]–[35] can search the structure of expression tree coding, and by virtue of its idea, the optimization mechanism can be designed to construct the hybrid model. The specific steps are as follows.

step1: Initialization. The population is composed of NP randomly generated I-ET coding individuals, which is expressed as $pop = \{\psi_1, \ldots, \psi_{NP}\}, j = 1, \ldots, NP$. Constructing ψ_j needs some initial parameters, such as the maximal depth D of the model, the function set F, the terminal set T, and the initial node depth of the model is 1. The specific constructing process is as follows.

1) If the depth of the current node is less than D, an element is randomly selected from $F \cup T$. Otherwise, do the same from T. The selected element is taken as *sp* of the current node and associated with a randomly generated *mp*. If *sp* belongs to F, turn to 2), otherwise turn to 3).

2) Identify the corresponding number of branches according to the number of children nodes of *sp* of the current node. For example, if *sp* is $+$, the number of child nodes is 2. The depth of the current node is plus 1, and returns to 1) to construct children nodes.

3) If *sp* of the current node belongs to variable in T, this variable will be set as the Gaussian node by the probability P_{gs}, and added what the Gaussian node calls attributes of the mean and variance that are randomly generated within a certain attribute range. The node as the terminal node of the branch, that is, the branch stops growing.

step2: Iterative search. In this process, the search operator probability P is used to choose the crossover or mutation operator and to generate the offspring individuals.

step2.1: Crossover operation. It can be expressed as $\{o_1, o_2\} = \psi_1 \otimes \psi_2$, in which $\otimes$ is the crossover operator, ψ_1 and ψ_2 are the two parent individuals randomly selected from the population, o_1 and o_2 are the two offspring individuals produced by crossover. The overall process of crossover is shown in Figure 4. Here are the concrete steps of crossover ψ_1 and ψ_2.

1) Select the crossover points of ψ_1 and ψ_2 separately according to the number of nodes of the models that generate numbers randomly.

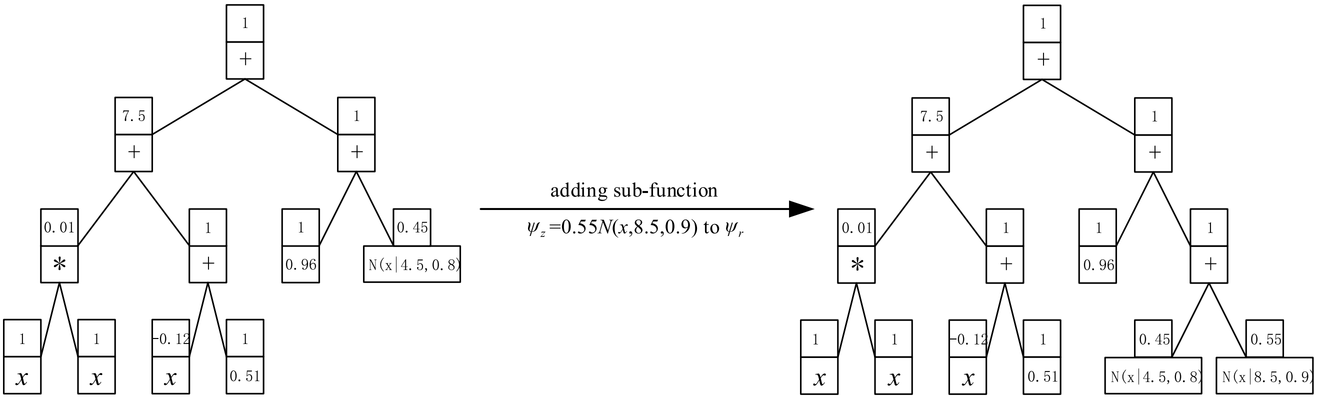

FIGURE 3. Coding structure before and after adding subfunction.

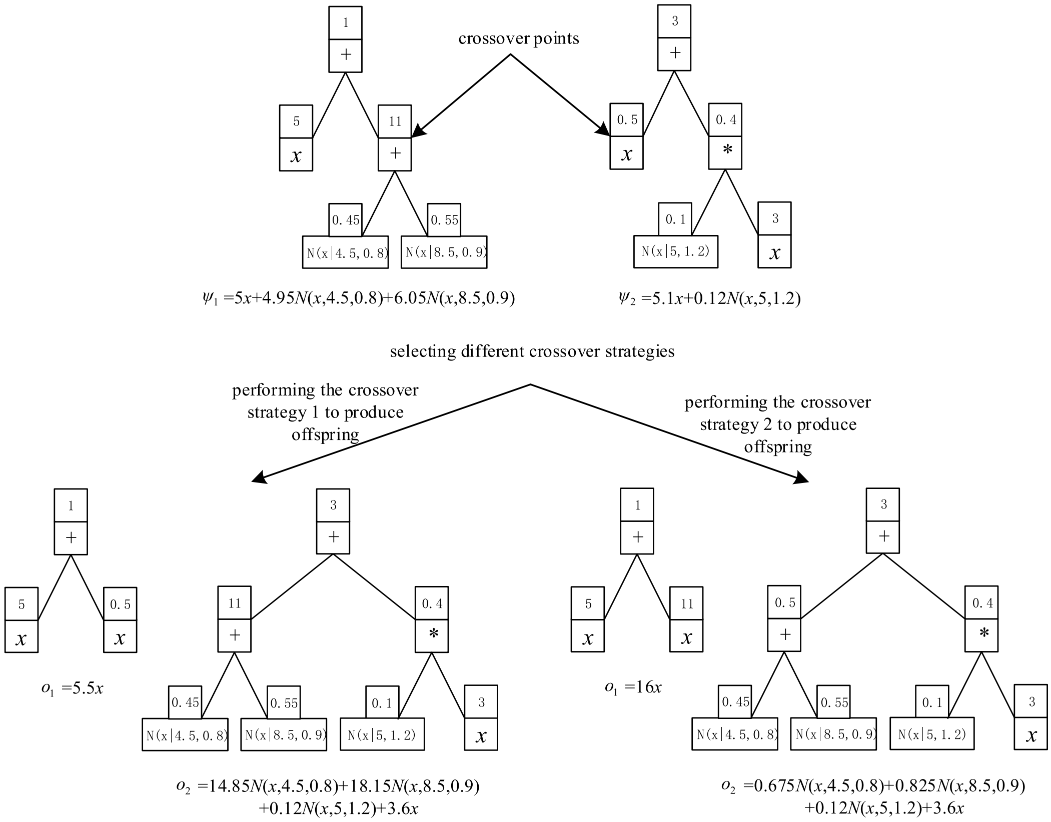
crossover points
ψ₁ =5x+4.95N(x,4.5,0.8)+6.05N(x,8.5,0.9)
ψ₂ =5.1x+0.12N(x,5,1.2)
selecting different crossover strategies
performing the crossover strategy 1 to produce offspring
performing the crossover strategy 2 to produce offspring
o₁ =5.5x
o₂ =14.85N(x,4.5,0.8)+18.15N(x,8.5,0.9) +0.12N(x,5,1.2)+3.6x
o₁ =16x
o₂ =0.675N(x,4.5,0.8)+0.825N(x,8.5,0.9) +0.12N(x,5,1.2)+3.6x
N(x|4. 5, 0. 8)
N(x|8. 5, 0. 9)
N(x|5, 1. 2)

FIGURE 4. Crossover operation of models.

2) According to the probability P_{cs}, choose from two different crossover strategies to do crossover operation. Figure 4 demonstrates comparison between the two crossover operations. The crossover strategy 1 takes *sp* and *mp* of the crossover point as a whole, and directly exchanges subtrees of the two parents with the crossover point as the root node. In the crossover strategy 2, the *mp* of the two crossover points is exchanged first, and then the subtrees with the crossover points as the root nodes are done likewise. In other words, *mp* is not exchanged with *sp*.

Obviously, the information processing granularity of the two crossover strategies is different. The first exchanges information with subfunctions as the unit, aiming to search for different subfunction combinations, while the second does likewise with the structure of subfunctions as the unit without relevant parameters, so as to keep the multiplier factor of the node changing after initializing the model.

step2.2: Mutation operation. It can be defined as $o = \odot(\psi)$, in which $\odot$ is the mutation operator, ψ is the parent individual selected randomly from the population, o is

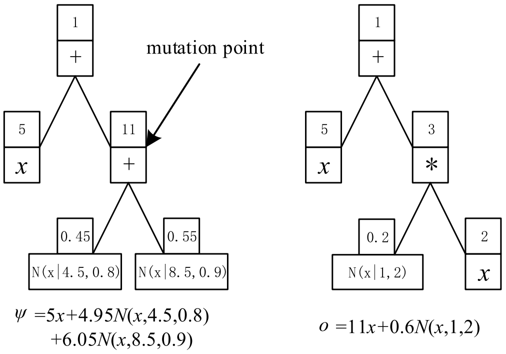

FIGURE 5. Mutation operation in the non-Gaussian node.

the offspring individual produced by mutation. The specific mutation process of ψ is as follows. First, select randomly mutation point of ψ. Second, since the attribute value of the Gaussian node determines particularity of the hybrid model, the mutation operation will be performed in use of different mutation strategies according to whether it is the Gaussian node.

When the mutation point is a non-Gaussian node, as is shown in Figure 5, delete the subtree whose root node is the mutation point, and then a new randomly generated subtree will be inserted.

When the mutation point is a Gaussian node, there are four ways of mutation: 1) mutation of the entire Gaussian node, namely, the mean, variance and *mp*. 2) mutation of the *mp*. 3) mutation of the mean. 4) mutation of the variance.

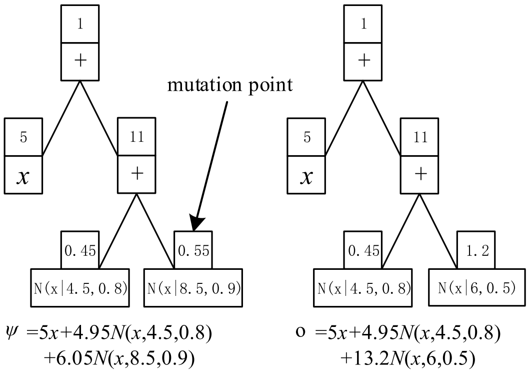

FIGURE 6. Mutation operation in the Gaussian node.

One of them will be randomly selected to perform mutation operation. Figure 6 shows the process of such operation in the first way of mutation.

step3: Model evaluation and selection. The offspring individual generated by searching will compete with the parent ones, and the superior will be selected to form the next generation population pop^{gen+1}. Model error f_{error} and complexity f_{node} are used to evaluate model. The former adopts the calculation formula in the *Definition 2*, while the latter uses the number of nodes of I-ET coding. Apparently, this is a process of bi-objective optimization.

When the two competing individuals ψ_1 and ψ_2 make following work, as shown in Equation 3

$$\begin{cases} f_{error}(\psi_1) \le f_{error}(\psi_2) \\ f_{node}(\psi_1) \le f_{node}(\psi_2), \end{cases} \tag{3}$$

when ψ_1 dominating ψ_2, ψ_2 will be eliminated. Yet when ψ_1 and ψ_2 do not control each other, the fitness will be used for comparison. The calculation formula of fitness *fit* is as shown in Equation 4

$$fit = \alpha f_{error}^{sf} + (1-\alpha) f_{node}^{sf}, \tag{4}$$

where α is proportion of adjusting two objectives, f_{error}^{sf} and f_{node}^{sf} are the normalized value. The normalization formula can be displayed as

$$f_{error}^{sf} = \frac{f_{error}(\psi) - f_{error}^{min}}{f_{error}^{max} - f_{error}^{min}}, \tag{5}$$

$$f_{node}^{sf} = \frac{f_{node}(\psi) - f_{node}^{min}}{f_{node}^{max} - f_{node}^{min}}, \tag{6}$$

where f_{error}^{min} and f_{error}^{max} are respectively the minimum and maximum of f_{error} in the current population, f_{node}^{min} and f_{node}^{max} are the minimum and maximum of f_{node}.

step4: If the number of individuals in the next population pop^{gen+1} is less than *NP*, turn to *step*2 to continue searching, otherwise, turn to *step5*.

step5: The current number of iterations *gen* plus 1. Determine whether the *gen* is the maximum. If so, output the best model. Otherwise, go to *step2*.

D. TIME COMPLEXITY ANALYSIS FOR PROPOSED METHOD

Suppose the population size is *NP*, and the maximal number of generations is *NG* during the execution of the algorithm.

In the initialization, the core operation is to randomly select nodes from function set and terminal set. So, the time complexity of initializing model ψ using I-ET coding is $O(N)$, N is the number of nodes of model, and the time complexity of *step1* is $O(NP \times N)$. In *step2.1*, the time complexity of selecting the crossover individuals is *O*(*NP*), and finding the crossover point by traversing tree is O(N), so the time complexity of *step2.1* is $O(NP \times N)$. In *step2.2*, the time complexity of selecting the mutation individual is *O*(*NP*), finding the mutation point by traversing tree is $O(N)$, so the time complexity of *step2.2* is $O(NP \times N)$. Therefore, the time complexity of *step2* is $O(NP \times N)$. In *step3*, the time complexity of calculating model complexity f_{node} is $O(NP \times N)$, and model error f_{error} is $O(NP \times N \times n)$, n is the number of sample points, so the time complexity of *step3* is $O(NP \times N \times n)$.

Overall, the time complexity of proposed algorithm is $O(NG \times NP \times N \times n)$. As the cost of building a good effort data fitting model, it is at an acceptable level.

IV. EXPERIMENTS

The proposed method was implemented on PC (2.3 GHz, 8 GB RAM, Windows 10) with MATLAB 2018a. In order

TABLE 2. Parameters setting.

Parameters	Value
Run times	10
Population size (NP)	50
Maximal number of generations (NG)	200
Maximal tree depth (D)	15
Function set (F)	{+, −, *,exp, ln, power}
Search operator rate(P)	self-adaptive [36]
Gaussian node rate (P_{gs})	0.5
Crossover strategies rate (P_{cs})	0.6
Weight (α)	0.7

TABLE 3. Information of data sets.

Data sets	Instances	The number of input variables	Output variables
Yacht Hydrodynamics	308	6	Residual resistance
Cooling efficiency	768	8	The cooling load
Heating efficiency	768	8	The heating load
Concrete	1030	8	Compressive strength
White wine quality	4898	11	White wine quality
Red wine quality	1599	11	Red wine quality

to verify the performance of proposed method, six problems of data prediction were selected for experimental study. Table 2 shows the relevant parameters setting of the method in the process of optimization. D is used to limit the infinite growth of the tree, P is used to choose the crossover or mutation operator, P_{cs} is used to choose crossover strategy, P_{gs} is used to generate gaussian node, and α is used to adjust f_{error} and f_{node}.

A. DATA SETS

There are six data sets in UCI machine learning database taken as test cases. Among them, the data set Hydrodynamics contains 308 instances, each of which is represented by seven attributes. In order to evaluate the ship's performance, it is great value to predict residual resistance of a ship at the beginning of design. The data set Energy efficiency contains data corresponding to 768 building shapes in description of eight attributes ranging from the surface area and the overall height. The purpose is to establish the relationship between the heating or cooling load and the above eight attributes. The data set Concrete contains data of 1030 different concrete samples described by eight attributes, such as cement, water and fly ash, aiming at identifying the relation between compressive strength of concrete and eight attributes. The last two data sets, White wine quality and Red wine quality, contain nearly 1599 and 4998 kinds of red and white wine samples respectively, with the aim to build a model that predicts quality of wine based on its 11 physical and chemical features. The specific information of the above data sets is shown in Table 3.

B. EVALUATION METRICS

This paper adopts 5-fold cross validation to evaluate the performance of the proposed method. The algorithm will run independently for 10 times, and calculate the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) between the predicted and the actual value. The final result is the average of the values obtained for all runs. MAE adopts f_{error} in *Definition 2* for calculation, and here is the calculation formula of RMSE

$$e_{\mathrm{RMSE}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i' - Y_i\right)^2}, \tag{7}$$

where Y_i is the observation value of the ith sample point and Y_i' is the prediction value, n is the number of the sample points in the test set which covers part of the sample points in the data set S.

C. COMPARISON OF EXPERIMENTAL RESULTS

This paper selects 7 comparison algorithms including the classical ones SVR, RF and XGBoost, and several improved methods proposed in recent years like ALAMO in the reference [37], OPLRA in [23] and the two approaches named PROA and PROB in [24]. The classical methods in the experiment use the sklearn algorithm package under Python 3.7. The main parameters are as follows: The kernel of SVR is rbf, the number of decision trees used in both RF and XGBoost is 200; the learning rate of XGBoost is 0.1. The results of other methods are directly adduced from the original references. Table 4 and Table 5 respectively present the average results of MAE and RMSE of each method.

TABLE 4. Comparison of MAE results of each method on the 6 data sets.

	Hydro	Cooling	Heating	Concrete	White	Red
SVR	3.673	2.455	2.682	8.195	0.634	0.567
RF	0.603	1.435	1.082	4.074	0.567	0.490
XGBoost	0.494	**1.096**	0.953	3.947	0.541	0.479
ALAMO	0.787	2.765	2.722	8.044	0.639	0.594
OPLRA	0.706	1.278	0.810	4.870	0.551	0.481
PROA	0.678	1.275	0.806	4.838	0.555	/
PROB	0.688	1.351	0.906	4.920	0.566	/
Proposed	**0.465**	2.0272	**0.693**	**3.8598**	**0.433**	**0.355**

TABLE 5. Comparison of RMSE results of each method on the 6 data sets.

	Hydro	Cooling	Heating	Concrete	White	Red
SVR	6.650	3.368	3.745	10.946	0.832	0.753
RF	1.238	2.100	1.480	5.495	0.714	0.627
XGBoost	0.976	**1.649**	1.682	5.445	0.691	0.626
ALAMO	/	/	/	/	/	/
OPLRA	1.402	2.022	1.507	6.883	0.771	/
PROA	1.207	1.989	1.508	6.811	0.778	/
PROB	1.226	2.079	1.619	6.885	0.782	/
Proposed	**0.841**	2.7450	**0.943**	**4.983**	**0.585**	**0.568**

Firstly, compare the proposed method in Table 4 with the three classical ones, SVR, RF and XGBoost. Of the three classical, XGBoost gets the minimum MAE. The proposed method is superior to SVR and RF in the six data sets, and works better than XGBoost in five sets except in the set Cooling. Then, the proposed performs better except in Cooling than those in the references like ALAMO, OPLRA, PROA and PROB. However, the proposed is inferior to OPLRA, PROA and PROB in dealing with the set Cooling.

All in all, the proposed gets the minimum MAE in five data sets but XGBoost has the minimum in Cooling.

According to the RMSE in Table 5, the performance of PROA is better than those in other references, so is that of RF which surpasses PROA in Cooling, Concrete and White. Besides, XGBoost also does better than PROA in four data sets except in Heating and Red. It turns out that the proposed method is apparently more advisable on the RMSE than PROA and XGBoost.

In order to evaluate the performance of each method for comparison in a more comprehensive way, the following scoring strategy is adopted. For each data set, arrange various methods according to their MAE and RMSE. The method with the lowest prediction error scores 10 points, the one that follows gets 9 points, and so on. For lack of partial results, the score of each method is the scores on average of the five data sets except Red. In addition, in use of RMSE scoring, ALAMO is excluded from calculating scores. The final average score represents the overall performance of the method. The higher score means the method works better. The average score of different methods is shown in Figure 7.

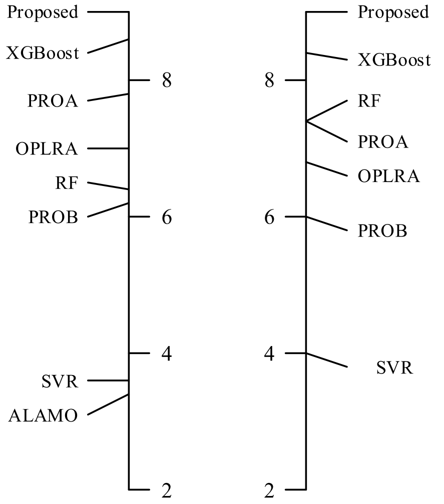

FIGURE 7. Average score of each method in MAE (left) and RMSE (right).

Figure 7 can make it easier to compare the performance of these methods. The proposed method gets the highest score in use of MAE and RMSE. XGBoost and PROA also work well. In addition, there are some differences when using MAE and RMSE for scoring. When RMSE is taken as the scoring indicator, RF is very competitive and has the same score as PROA.

This paper also adopts the Welch t test to compare the significance of differences in performance of each method at the significance level of 5%. The MAE values of these methods on each data set are used. The test results are shown in Table 6, in which the "+" indicates that the proposed method is obviously superior to the comparison one; the "≈" means no significant difference between the two methods; the "-" suggests that the proposed is largely inferior to the other.

TABLE 6. Results of the statistical significance test.

Data sets	The proposed method		
	SVR	RF	XGBoost
Hydro	+	+	≈
Cooling	+	–	–
Heating	+	+	+
Concrete	+	≈	≈
White	+	+	+
Red	+	+	+

As seen from Table 6, when compared with SVR, the proposed method is superior in the six data sets. In comparison with RF, the proposed works worse in Cooling set, there is no significant difference between the two in Concrete set, but the proposed is superior to RF in the rest of the four data sets. When compared with XGBoost, the proposed is inferior in Cooling set, similar performance of both approaches is presented in Concrete and Hydro sets, yet the proposed is better than XGBoost in the rest of the three sets.

TABLE 7. Comparison of two coding mechanisms.

Data Sets	Coding	train f_{error}	test f_{error}	avg f_{node}
Hydro	traditional	0.867	0.857	37.848
	I-ET	0.870	0.483	18.532
Cooling	traditional	3.238	2.682	43.161
	I-ET	2.194	2.173	29.583
Heating	traditional	2.482	2.678	57.645
	I-ET	1.292	0.721	36.543
Concrete	traditional	8.142	6.094	67.347
	I-ET	3.637	3.849	45.709
White	traditional	0.513	0.608	69.672
	I-ET	0.498	0.446	47.862
Red	traditional	0.576	0.502	57.134
	I-ET	0.525	0.387	37.327

D. COMPARISON OF I-ET CODING AND TRADITIONAL CODING

In order to evaluate the influence of the I-ET coding and traditional tree coding on the model performance, the training error (train f_{error}), the test error (test f_{error}), the average complexity of the population in the whole optimization process(avg f_{node}) and the running time cost are compared respectively for 6 test problems. The two coding mechanisms run ten times respectively under the same hardware, software environment, and stop condition. Table 7 shows the comparison of train f_{error}, test f_{error} and avg f_{node} between them. Figure 8 shows the variation of running time cost and avg f_{node} on 6 data sets. Figure 9 displays the curve of average complexity of the population of the two coding mechanisms throughout the optimization process in Hydro.

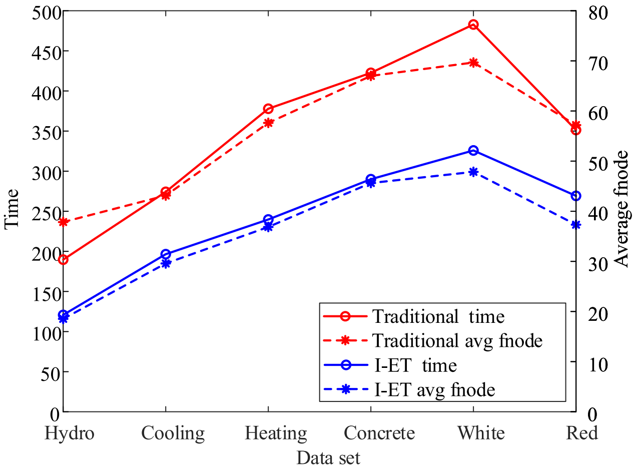

FIGURE 8. **Variation of running time and avg fnode of two coding mechanisms on 6 data sets.**

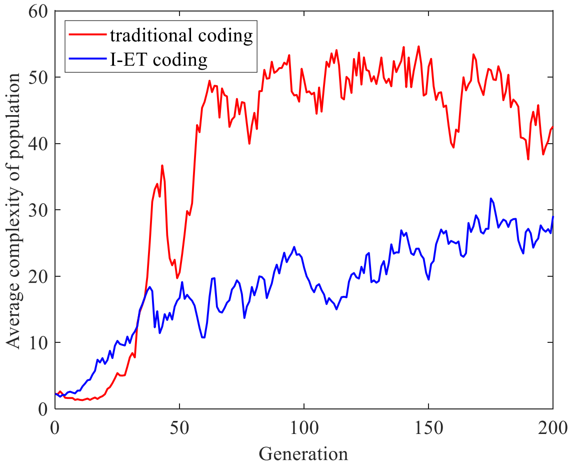

FIGURE 9. **Variation curve of average complexity of population in Hydro.**

It can be seen from Table 7 that the I-ET coding is superior to the traditional one in three indicators on all data sets. Compared with the traditional coding, the train f_{error}, the test f_{error} and the avg f_{node} are reduced by an average of 24.5%, 37.0%, and 36.2% respectively. Thus, the I-ET being more effective in accuracy and complexity. Meanwhile, it can be seen from Figure 8 that the time cost has a strong correlation with the avg f_{node} of population. The larger avg f_{node} is, the longer time cost is. Figure 9 can further reflect that the I-ET coding can reduce the model complexity compared to the traditional coding. In summary, the I-ET coding reduce the model complexity and time cost distinctly while decrease the train and test f_{error} visibly.

E. INTERPRETABILITY OF THE HYBRID MODEL

Taking the Hydro set as example, the interpretability of the proposed and polynomial model is analysed. The expression of the 3-order polynomial model y_p is as shown in Equation 8, as shown at the bottom of this page, and the expression of the hybrid model ψ by evolutionary search in this paper is as shown in Equation 9, as shown at the bottom of this page.

The expressions of the two models suggest that the 3- order polynomial model y_p is very huge and complex, of which

$$\begin{aligned}
y_p = {} & 439687704.88x_1 + 692566197.05x_2 - 57033897.31x_3 - 2162836122.90x_4 + 2586958939.04x_5 + 20524.98x_6 \\
& -1444014141.72x_1^2 - 886183799.29x_1x_2 + 738595195.50x_1x_3 + 3396306733.15x_1x_4 - 926009293.46x_1x_5 \\
& -187.18x_1x_6 + 100655337.67x_2^2 + 1475654392.31x_2x_3 + 1219775641.45x_2x_4 + 402020241.07x_2x_5 - 71146.47 \\
& {}^*x_2x_6 - 7174034771.43x_3^2 - 2850387830.08x_3x_4 + 4789985237.27x_3x_5 - 17110.99x_3x_6 + 5201024276.52x_4^2 \\
& +361449352.60x_4x_5 + 6734.76x_4x_6 + 3953128150.62x_5^2 + 17032.64x_5x_6 - 653.95x_6^2 - 23350865.67x_1^3 \\
& -597825366.91x_1^2x_2 + 538126753.47x_1^2x_3 + 134851277.83x_1^2x_4 - 474374683.70x_1^2x_5 + 1.13x_1^2x_6 + 323755249.90 \\
& {}^*x_1x_2^2 - 2447226091.17x_1x_2x_3 + 822526498.89x_1x_2x_4 + 742180122.41x_1x_2x_5 + 2784.07x_1x_2x_6 + 1726594305.45 \\
& {}^*x_1x_3^2 - 5607934878.14x_1x_3x_4 - 605805168.08x_1x_3x_5 + 192.85x_1x_3x_6 + 1687868165.76x_1x_4^2 + 3380348448.52 \\
& {}^*x_1x_4x_5 - 392.71x_1x_4x_6 - 388917383.95x_1x_5^2 - 234.53x_1x_5x_6 + 13.74x_1x_6^2 + 219619865.35x_2^3 - 154147640.97x_2^2x_3 \\
& -890414756.84x_2^2x_4 - 654231614.30x_2^2x_5 + 76935.00x_2^2x_6 + 1293675619.57x_2x_3^2 + 1413972161.47x_2x_3x_4 \\
& +2217624846.02x_2x_3x_5 + 29513.91x_2x_3x_6 - 1848766379.19x_2x_4^2 - 2916542598.42x_2x_4x_5 - 13476.15x_2x_4x_6 \\
& -4225869702.08x_2x_5^2 - 30038.80x_2x_5x_6 - 3558.12x_2x_6^2 + 4314944403.57x_3^3 - 6553921887.10x_3^2x_4 - 1017900157.65 \\
& {}^*x_3^2x_5 + 2864.10x_3^2x_6 + 3083162770.37x_3x_4^2 + 4283295181.66x_3x_4x_5 - 2253.86x_3x_4x_6 - 8990511074.04x_3x_5^2 \\
& -5529.43x_3x_5x_6 - 40.03x_3x_6^2 - 574826647.71x_4^3 - 1907684370.43x_4^2x_5 + 449.89x_4^2x_6 + 2736060535.80x_4x_5^2 \\
& +2265.46x_4x_5x_6 - 37.12x_4x_6^2 + 5265571773.39x_5^3 + 2655.48x_5^2x_6 - 8.33x_5x_6^2 + 4532.00x_6^3 + 17780636561.72,
\end{aligned} \tag{8}$$

$$\begin{aligned}
\psi = {} & 0.032e^{10.091x_6+0.943N(x_6,0.385,0.023)} + 0.568\ln(3.126x_6) + 0.121N(x_3, 4.691, 0.209) + 0.074N(x_4, 3.486, 0.629) \\
& +0.141N(x_6, 0.438, 0.859) + 0.117N(x_6, 0.385, 0.023) + 0.169N(x_1, -4.65, 0.481) + 0.145N(x_1, -0.098, 0.827) \\
& +0.027N(x_1, -4.616, 0.342) + 0.086N(x_3, 4.851, 0.073) + 1.34x_6,
\end{aligned} \tag{9}$$

prediction error is 1.406. That of the 4-order polynomial model is 0.643, which is not listed here since the expressions are too complicated. Compared with y_p, the hybrid model ψ generated by the proposed method has simpler structure and smaller error of only 0.483. It is more accurate in prediction than the 3-order and 4-order polynomial models as well.

In addition, simplification of the hybrid model structure makes analysis of the prediction results more convenient. For example, in the results of the Hydro set, the model constructed by the proposed method highlights the importance of the attributes x_1, x_3, x_4 and x_6, which respectively correspond to the longitudinal position of the center of buoyancy, length-displacement ratio, beam-draught ratio and Froude number. Obviously, on the basis of these results, it is more convenient to further analyze which attributes or their combinations that have a greater impact on residual resistance of the ship. Therefore, the proposed method can make the model much less complex and allow further interpretation.

F. PARAMETERS DISCUSSION

Take the set Hydro as example once again to analyze four major parameters.

1) INFLUENCE OF WEIGHT ON MODEL ERROR AND COMPLEXITY

When fitness of the model is calculated, the weight α is an important parameter affecting model error and complexity, and determines the evolutionary direction of the population.

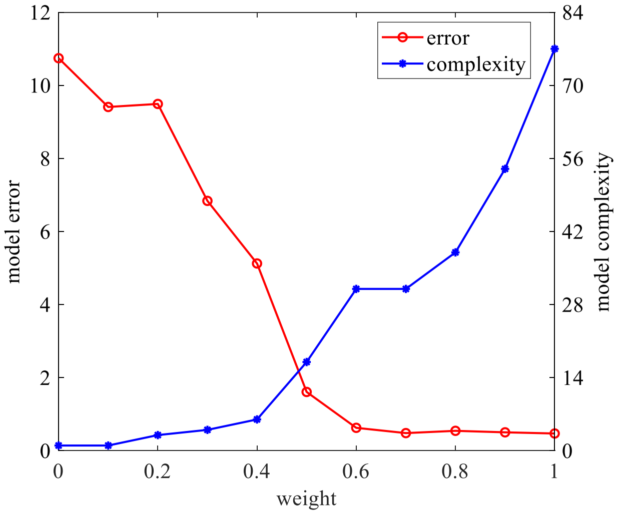

FIGURE 10. Influence of different weight on model performance.

If α is too small, the complexity of the model will be the dominant factor, and the whole population will evolve in the direction of less complexity. But as the model with low complexity contains too little information, the model might be less accurate. In contrast, if α is too large, reducing the model error will lead the drive of evolution, but the model is more likely to be rather complicated. Figure 10 shows the influence of different values of α on model error and complexity. It can be seen that when α is within 0.6 to 0.8, both accuracy and simplicity of the model will be guaranteed.

2) INFLUENCE OF PROBABILITY P OF SEARCH OPERATOR ON MODEL ERROR

In the search process, the related genetic operation to produce offspring is selected based on the probability P, which is the key parameter influencing the performance of GP. The larger P results in the greater probability of choosing crossover operation and faster generation of new individuals. But the structure of excellent individuals will be destroyed quickly. In contrast, the smaller P is, the greater the probability of choosing mutation operation is, so the searching process will become random. When it comes to the state of the population in the evolutionary process, calculating P under a self-adaptive mechanism makes dynamic variability of the population adaptable. Figure 11 shows the influence of different calculation strategies of P on model error in the iteration process. It can be seen that the model error curve with the fixed value up to 0.9 of P is higher than that with the value of P obtained from a self-adaptive mechanism. When this mechanism is used to determine the probability of the search operator, selecting different genetic operations can improve the searching performance of algorithms.

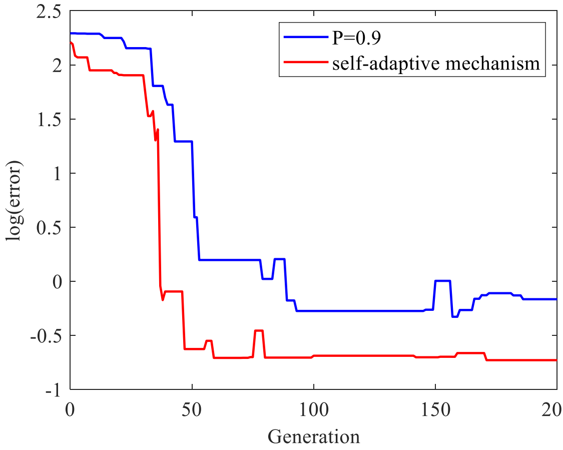

FIGURE 11. Error curve of *P* under different mechanisms.

3) INFLUENCE OF PROBABILITY P_{cs} OF DIFFERENT CROSSOVER STRATEGIES ON MODEL ERROR

When the crossover strategy is selected, the probability P_{cs} can affect model error. If P_{cs} is too small, the search particle will be too large, but if it is too large, the combination of searching subfunctions will be at the very heart, and the search of relevant parameters is little. Figure 12 reflects the influence of P_{cs} on model error, suggesting that with the increase of P_{cs}, the model error decreases and then increases. When P_{cs} is between 0.5 and 0.7, the two crossover strategies can be well balanced.

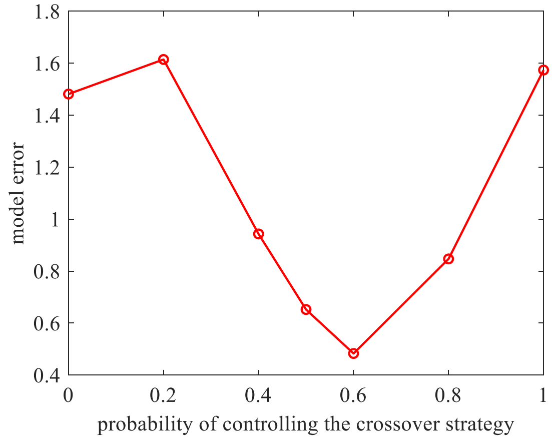

FIGURE 12. Influence of P_{cs} on model error.

4) INFLUENCE OF PROBABILITY P_{gs} OF GENERATING GAUSSIAN NODE ON MODEL ERROR

When individuals are initialized or mutated, the probability P_{gs} turning from ordinary variable to the Gaussian node will influence model error which can be reduced thanks to the Gaussian function. If P_{gs} is too small, the Gaussian function is less likely to appear in the hybrid model. As a result, there is great model error. But if P_{gs} is too large, the resulting superfluous Gaussian function in the model and insufficient information about other models will cause larger model error likewise.

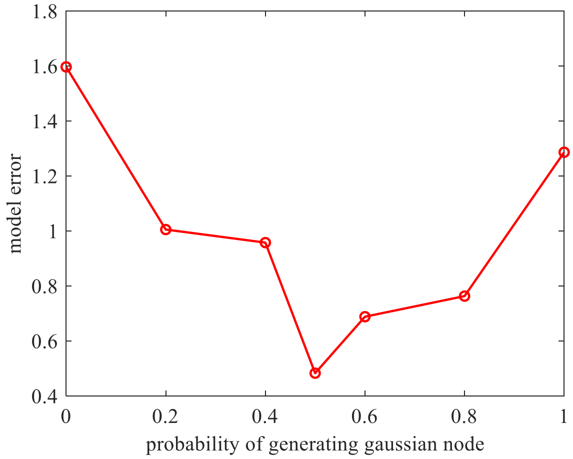

FIGURE 13. Influence of P_{gs} on model error.

Figure 13 demonstrates the influence of different P_{gs} on model error. When P_{gs} grows, overall model error decreases and then increases. And when P_{gs} is 0.5, model error is the lowest and it also performs well.

V. CONCLUSION

This paper presents the method of constructing a hybrid data fitting model based on I-ET coding and evolutionary search. This approach is proven to be effective by experiments on the six UCI data sets, and comparison of the results by different means. The results suggest that the proposed method for model construction can bring forth hybrid model with higher prediction accuracy and lower complexity. Meanwhile, this paper also discusses how I-ET coding adopted in the proposed method makes calculation less complex and the relationship between the selection of four important parameters and the performance of the constructed model.

In future work, the following work will be continued. Firstly, the effectiveness of the proposed approach was verified only on UCI datasets, it is a general framework that can be applied to predict other practical problems. Then, the coding of hybrid model is the fundamental problem, which directly affects the efficiency of the whole algorithm. Although I-ET coding can reduce the complexity of model, it is still complex in the search process. The efficient and simple model coding structure can be further explored to improve the efficiency of search operation. Finally, the multi-objective optimization technique needs to be further studied, so as to construct the hybrid model with better performance.

REFERENCES

[1] F. Karimi, S. Sultana, A. S. Babakan, and S. Suthaharan, "An enhanced support vector machine model for urban expansion prediction," *Comput., Environ. Urban Syst.*, vol. 75, pp. 61–75, May 2019.

[2] V. Sousa, I. Meireles, V. Oliveira, and C. Dias-Ferreira, "Prediction performance of separate collection of packaging waste yields using genetic algorithm optimized support vector machines," *Waste Biomass Valorization*, vol. 10, no. 12, pp. 3603–3612, Dec. 2019.

[3] Y. Y. Cheng, "Short-term electricity demand forecasting based on artificial neural network," M.S. thesis, Dept. Elect. Eng., Zhejiang Univ., Hangzhou, China, 2017.

[4] Y. Lee, D. Han, M.-H. Ahn, J. Im, and S. J. Lee "Retrieval of total precipitable water from Himawari-8 AHI data: A comparison of random forest, extreme gradient boosting, and deep neural network," *Remote Sens.*, vol. 11, no. 15, pp. 1–19, 2019.

[5] K. Lin, Y. Hu, and G. Kong, "Predicting in-hospital mortality of patients with acute kidney injury in the ICU using random forest model," *Int. J. Med. Informat.*, vol. 125, pp. 55–61, May 2019.

[6] I. A. Ibrahim, M. J. Hossain, and B. C. Duck, "An optimized offline random forests-based model for ultra-short-term prediction of PV characteristics," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 202–214, Jan. 2020.

[7] M. Al-Rakhami, A. Gumaei, A. Alsanad, A. Alamri, and M. M. Hassan, "An ensemble learning approach for accurate energy load prediction in residential buildings," *IEEE Access*, vol. 7, pp. 48328–48338, 2019.

[8] S. Ji, X. Wang, W. Zhao, and D. Guo, "An application of a three-stage XGBoost-based model to sales forecasting of a cross-border E-commerce enterprise," *Math. Problems Eng.*, vol. 2019, pp. 1–15, Sep. 2019.

[9] H. N. Akouemo and R. J. Povinelli, "Data improving in time series using ARX and ANN models," *IEEE Trans. Power Syst.*, vol. 32, no. 5, pp. 3352–3359, Sep. 2017.

[10] H.-I. Suk, C.-Y. Wee, S.-W. Lee, and D. Shen, "State-space model with deep learning for functional dynamics estimation in resting-state fMRI," *NeuroImage*, vol. 129, pp. 292–307, Apr. 2016.

[11] E. Hoseinzade and S. Haratizadeh, "CNNpred: CNN-based stock market prediction using a diverse set of variables," *Expert Syst. Appl.*, vol. 129, pp. 273–285, Sep. 2019.

[12] W. Xu, H. Peng, X. Zeng, F. Zhou, X. Tian, and X. Peng, "A hybrid modelling method for time series forecasting based on a linear regression model and deep learning," *Int. J. Speech Technol.*, vol. 49, no. 8, pp. 3002–3015, Aug. 2019.

[13] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[14] W. Kong, Z. Y. Dong, D. J. Hill, F. Luo, and Y. Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1087–1088, Jan. 2018.

[15] R. S. Arranz and A. Gutiérrez, "A long short-term memory artificial neural network to predict daily HVAC consumption in buildings," *Energ. Buildings*, vol. 216, no. 6, pp. 1–13, 2020.

[16] C. Y. Yang, H. Li, and Z. Y. Che, "Energy consumption modeling and parameter identification for double motor driven coal mine belt conveyers," *Control Theory Appl.*, vol. 35, no. 3, pp. 335–341, 2018.

[17] B. Cheng, X. F. Qiu, L. Y. Ji, L. P. Meng, R. H. Li, H. D. Wang, and Z. Q. Xu, "Modeling and application of BDS satellite clock error," *Sci. Surv. Mapp.*, vol. 44, no. 10, pp. 14–20, 2019.

[18] X. L. Jiang, K.-P. Feng, H. Lin, J. Tang, Z.-Z. Zhou, and J. Li, "Active contours driven by local Gaussian distribution fitting and local robust statistics," *J. Inf. Hiding Multimeda Signal Process.*, vol. 9, no. 1, pp. 89–98, 2018.

[19] S. Xie, Y. F. Huang, T. J. Li, C. Y. Liu, and J. H. Wang, "Mid long-term runoff prediction based on a lasso and SVR hybrid method," *J. Basic Sci. Engine.*, vol. 26, no. 4, pp. 709–722, 2018.

[20] G. Hesamian and M. G. Akbari, "Fuzzy lasso regression model with exact explanatory variables and fuzzy responses," *Int. J. Approx. Reasoning*, vol. 115, pp. 290–300, Dec. 2019.

[21] F. Ziel, C. Croonenbroeck, and D. Ambach, "Forecasting wind power—Modeling periodic and non-linear effects under conditional heteroscedasticity," *Appl. Energy*, vol. 177, pp. 285–297, Sep. 2016.

[22] Y. He, Y. Qin, S. Wang, X. Wang, and C. Wang, "Electricity consumption probability density forecasting method based on LASSO-quantile regression neural network," *Appl. Energy*, vols. 233–234, pp. 565–575, Jan. 2019.

[23] L. Yang, S. Liu, S. Tsoka, and L. G. Papageorgiou, "Mathematical programming for piecewise linear regression analysis," *Expert Syst. Appl.*, vol. 44, pp. 156–167, Feb. 2016.

[24] I. Gkioulekas and L. G. Papageorgiou, "Piecewise regression analysis through information criteria using mathematical programming," *Expert Syst. Appl.*, vol. 121, pp. 362–372, May 2019.

[25] Q. F. Wang, S. Zhang, Z. Chen, B. L. Zhang, and C. Jiang, "Thermal error compensation model of machine spindle based on exponential function," *Comput. Integr. Manuf. Syst.*, vol. 21, no. 6, pp. 1553–1558, 2015.

[26] Y. C. Xu, X. Wang, X. Yin, R. Yue, Z. Hu, and J. Sun, "Potato processing quality characteristics prediction based on multivariate nonlinear regression analysis," *Trans. Chines. Soc. Agricult. Mach.*, vol. 49, no. 4, pp. 366–373, 2018.

[27] A. G. Memon, R. A. Memon, K. Harijan, and M. A. Uqaili, "Parametric based thermo-environmental and exergoeconomic analyses of a combined cycle power plant with regression analysis and optimization," *Energy Convers. Manage.*, vol. 92, pp. 19–35, Mar. 2015.

[28] J. P. Wang, C. Y. Huang, Y. Liang, and L. B. Shao, "Thermal stress analysis and optimization of BGA solder joint power load based on regression analysis and genetic algorithm," *Acta Electron. Sinica*, vol. 47, no. 3, pp. 224–230, 2019.

[29] A. Shabri and R. Samsudin, "Crude oil price forecasting based on hybridizing wavelet multiple linear regression model, particle swarm optimization techniques, and principal component analysis," *Sci. World J.*, vol. 2014, pp. 1–8, May 2014.

[30] C. Su-Fen, "Dynamic population structure based PSO with granular computing for unified multiple linear regression," *Inf. Technol. J.*, vol. 12, no. 24, pp. 8430–8434, Dec. 2013.

[31] H. Sheng, J. Xiao, and P. Wang, "Lithium iron phosphate battery electric vehicle state-of-charge estimation based on evolutionary Gaussian mixture regression," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 544–551, Jan. 2017.

[32] A. Ramezani, A. R. Khan, B. Moshiri, and B. Abdulhai, "Design of an adaptive maximum likelihood estimator for key parameters in macroscopic traffic flow model based on expectation maximum algorithm," *IET Sci., Meas. Technol.*, vol. 5, no. 5, pp. 189–197, Sep. 2011.

[33] K. Özkan, Ş. Işik, Z. Günkaya, A. Özkan, and M. Banar, "A heating value estimation of refuse derived fuel using the genetic programming model," *Waste Manage.*, vol. 100, pp. 327–335, Dec. 2019.

[34] I. De Falco, A. D. Cioppa, A. Giugliano, A. Marcelli, T. Koutny, M. Krcma, U. Scafuri, and E. Tarantino, "A genetic programming-based regression for extrapolating a blood glucose-dynamics model from interstitial glucose measurements and their first derivatives," *Appl. Soft Comput.*, vol. 77, pp. 316–328, Apr. 2019.

[35] R. Vyas, S. Bapat, P. Goel, M. Karthikeyan, S. S. Tambe, and B. D. Kulkarni, "Application of genetic programming (GP) formalism for building disease predictive models from protein-protein interactions (PPI) data," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 1, pp. 27–37, Jan. 2018.

[36] Y. Xing, C. Chen, L. L. Liu, S. Cheng, and Y. N. Su, "Dam deformation prediction based on improved genetic algorithm and BP neural network," *Comput. Engine. Des.*, vol. 39, no. 8, pp. 2628–2631 and 2686, 2018.

[37] A. Cozad, N. V. Sahinidis, and D. C. Miller, "Learning surrogate models for simulation-based optimization," *AIChE J.*, vol. 60, no. 6, pp. 2211–2227, Jun. 2014.

HAO CHEN received the B.S. degree in computer science and the Ph.D. degree in power electronics and power transmission from the Xi'an University of Technology, Xi'an, China. He is currently an Associate Professor with the School of Computer Science and Technology, Xi'an University of Posts and Telecommunications. He has authored more than 40 articles. His current research interests include evolutionary algorithms and their applications in the real world.

ZI YUAN GUO is currently pursuing the master's degree in computer application technology with the Xi'an University of Posts and Telecommunications, China. Her main research interests include data mining and intelligent computing.

HONG BAI DUAN is currently pursuing the master's degree in big data processing and high-performance computing with the Xi'an University of Posts and Telecommunications, China. His main research interests include data mining and machine learning.

DUO BAN is currently pursuing the master's degree in computer technology with the Xi'an University of Posts and Telecommunications, China. His main research interests include data mining and machine computing.