Evolutionary modeling algorithm of ordinary differential equations based on genetic modeling

Zhiqin Chen*
Jiangxi Teachers College, Yingtan, Jiangxi 335000, China
*Corresponding author: chengzhiqing168@163.com

Abstract. Differential equations are often used to describe complex systems and nonlinear systems related to time, but it is difficult to establish an ideal model for such systems based on some observed data. Especially when dealing with unknown chaotic system data, it is blind and difficult to combine existing nonlinear analysis results with relevant experience. In this paper, through the study of genetic modeling, the optimization process of model parameters using GA is embedded in the optimization process of model structure using GP, and the local search process of neighborhood solution generated by GP-based standard mutation operator is carried out for some individuals in each evolution generation, and the evolution modeling algorithm of ordinary differential equations is designed and implemented, and an application example is given.

Keywords: Genetic modeling; Ordinary differential equation; Modeling; Genetic programming

1. Introduction

Traditional mathematical methods to solve the problem of formula discovery include curve fitting, regression analysis and approximation theory. However, it is very difficult to establish an ideal model for this kind of system based on some observation data. Traditional methods often require complex parameter selection process and sufficient modeling experience, and their adaptability is narrow [1]. Neural network and polynomial are two commonly used global modeling tools [2-3]. However, most of these global modeling methods can not give a concise and intuitive model expression, especially when dealing with unknown chaotic system data, which is blind and difficult to combine the existing nonlinear analysis results and related experiences. The established one-time model can not reflect the real-time characteristics of data, but in practical application, it is often necessary to constantly modify the structure and parameters of the model with the update of observation data in order to achieve the effect of real-time modeling and forecasting [4].

The basic structure and characteristics of evolutionary algorithm determine that it is especially suitable for large-scale parallelism. Bethke started the earliest research work on parallel genetic algorithm in 1976 [5-6]. Since then, people have studied various types of parallel GA according to different models, including global parallel GA, coarse-grained parallel GA, fine-grained parallel GA...
and hybrid parallel GA [7]. Literature [8] puts forward a model structure optimized by traditional genetic programming, which embeds the process of model parameter optimization by genetic algorithm during evolution, and can obtain a better model by adaptive evolution in a short time. The new technology combining genetic programming with local search is used to optimize the model structure. By applying the new algorithm to population modeling and chemical reaction modeling, the effectiveness of the new algorithm is demonstrated by comparing the modeling results of the models obtained by the old and new algorithms running many times.

2. Genetic modeling description

Genetic modeling is composed of function set and endpoint set of the problem, and its main operators are composed of selection operator, exchange operator and mutation operator. Traditional genetic operators inherit the idea of genetic algorithm. Each individual is given a score according to the calculation content, and the score is the main content to guide the environment. Because the complex performance of chaotic systems is often dominated by simple low-dimensional deterministic motion laws, and often includes positive feedback links, switching links, delay links or interaction links, etc. Different from the standard GP technology, each chromosome here is represented as a multi-tree structure to represent a differential equation system model; Accordingly, individual hybridization can be divided into individual level and tree level.

Operators are described as follows:

1. Selection operator: select the next generation with smaller error according to a certain selection algorithm in the group;
2. Exchange operator: select two parents according to probability, randomly select exchange points, exchange two subtrees on the exchange points, and join the next generation population;
3. Mutation operator: according to the probability, select the mutated individual, randomly select the mutation point for mutation, and add the mutated individual to the next iteration.

Formula discovery is defined as follows [9]:

\[ \text{Give a set of observation data } D = \{(X_i, y_i) | X_i \in R^n, y_i \in R, i = 1, 2, \ldots, N\} \]

The finding formula is:

\[ f : R^n \rightarrow R \quad (1) \]

\[ \sum |f(X_i) - y_i| \approx 0 \quad (2) \]

According to the definition of formula discovery, genetic modeling is applied to formula discovery, which is defined as follows:

Let \( \{(x, y), x \in X, y \in Y, i \in I\} \) be a given input-output pair, where \( X, Y \) is a subset in a finite dimensional space and \( I \) is an index set;

If \( F \) is a subset of \( C(X) \), \( C(X) \) is the whole of all continuous functions on \( X \), and \( \rho \) is the distance defined on the product space \( \prod Y \), then the modeling problem is to determine a function \( f^* \in F \) so that there is any \( f \in F \)

\[ \rho(\{f^*(x_i), \{y_i\}\}) \leq \rho(\{f(x_i), \{y_i\}\}) \quad (3) \]

Equation (3) holds, or makes for a given \( \varepsilon > 0 \)

\[ \rho(\{f^*(x_i), \{y_i\}\}) \leq \varepsilon \quad (4) \]

From the expressions (3) and (4), the modeling problem can be regarded as the following optimization problem

\[ \min_{f \in F} \rho(\{f(x_i), \{y_i\}\}) \quad (5) \]
3. Problem description and coding method

If a dynamic system can be represented by \( n \) time-related functions \( x_1(t), x_2(t), \ldots, x_n(t) \), and the data observed at \( m \) time points are:

\[
X = \begin{bmatrix}
x_1(t_0) & x_2(t_0) & \cdots & x_n(t_0) \\
x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\
\vdots & \vdots & \ddots & \vdots \\
x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m)
\end{bmatrix}
\]

(6)

Where \( t_0 \) is the starting time, \( t_i = t_0 + i \times \Delta t \), \( i = 0, 1, \ldots, m \), \( \Delta t \) is the time interval, and \( x_j(t_i) \) is the observation value of variable \( x_j \) at \( t_i \) time. Note:

\[
X(t) = [x_1(t), x_2(t), \ldots, x_n(t)]
\]

(7)

\[
f(t, X) = [f_1(t, X), f_2(t, X), \ldots, f_n(t, X)]
\]

(8)

In which \( f_j(t, X) = f_j(t, x_1(t), x_2(t), \ldots, x_n(t)) \), \( j = 1, 2, \ldots, n \) is represented by the compound function of elementary functions. Remember that the space composed of such functions is \( F \).

The modeling problem of ordinary differential equations is to find the function expression \( f(t, X^*) \) and satisfy it

\[
dX^*/dt = f(t, X^*)
\]

(9)

And minimize \( \|X^* - X\| \), where

\[
\|X^* - X\| = \sqrt{\sum_{i=0}^{m} \sum_{j=0}^{n} (x_j(t_i) - x_j(t_i))^2}
\]

(10)

According to the established model, we can forecast the \( X(t)(t > t_m) \) at the later time.

4. Hybrid evolutionary modeling algorithm

4.1 Hybrid GP technology

O'Reilly et al. have studied the hybrid algorithm combining genetic programming (GP) with other search techniques, and obtained some experimental results superior to the standard GP technique [10-11]. Inspired by this, in the process of optimizing the model structure, we use the "mountain climbing" technology to search for some new solutions obtained by GP genetic operation, in which the mutation operator in GP is used to generate neighborhood solutions. The process can be described in C-like language as follows:

GP+ L S _ MU algorithm: hybrid gp technology optimizes model structure

\{
\begin{align*}
&\text{for } (i = 0; i < MAX; i ++) \\
&\text{Randomly select the individual } P \text{ in } P(t); \text{ for } (k = 0; k < L; k ++) \\
&\text{Generating a variant solution } P^* \text{ of } P; \\
&\text{The adaptive value } \text{fitness} (P^*) \text{ of } P^* \text{ is calculated; } \\
&\text{if } (\text{fitness} (P^*) < \text{fitness} (P)) P = P^*; \}
\end{align*}
\}

Among them, \( P(t) \) is the model population of GP after genetic manipulation of \( t \) generations, \( MAX \) is the number of individuals selected for local search in each generation, and \( L \) is the maximum
number of individual variations.

4.2 GP+ GA algorithm
This kind of algorithm adopts the idea of two-level evolutionary modeling proposed in literature [12], optimizes the structure of the model by genetic programming and optimizes the parameters of the model by genetic algorithm. The difference is that a more effective genetic algorithm is used here, that is, genetic algorithm based on subspace search. It is simple, stable and efficient. Assuming that each equation of the ordinary differential equation system (10) contains \( l_i(i:1-n) \) parameters, each individual \( V \) in the parameter population can be expressed as a column vector form composed of \( n \) dimensional \( l_i \) row vectors.

\[
V = (C_1C_2\cdots C_n)^T \in D \subset R^k
\]  

(11)

In which \( C_i = (c_{i1}, c_{i2}, \cdots, c_{il_i}) \), \( D \) is a parameter space. Then \( M \) test points (vectors) in the parameter space can be expressed as \( V_1, V_2, \cdots, V_M \) (they are not necessarily linearly independent).

Assuming that \( M \) random coefficients \( \alpha(i:1-M) \) satisfy the conditions \( a \leq \alpha \leq b (a < 0, b > 1) \) and \( \sum_{i=1}^{M} \alpha = 1 \), and the subspace \( S = V = \sum_{i=1}^{M} \alpha V_i \) determined by the random nonconvex combination of \( M \) test points is set as \( F(x) \), the GA algorithm for parameter optimization can be described in class c language as follows:

**GA algorithm:**

\{ t = 0; 

Initialize parameter population \( Q(t) = (V_1(t), V_2(t), \cdots, V_N(t)) \);  

Among them \( V_i(t) (i:1 \sim N) \in \);  

Calculation \( F(Q(t)) = (F(V_1), F(V_2), \cdots, F(V_N)) \);  

while (Termination conditions are not met)  

\{ \( M \) different individuals \( V_i(i:1 \sim) \) are randomly selected from \( Q(t) \);  

Generate \( M \) random coefficients \( \alpha(i:1 \sim) \);  

Generate a new individual \( V = \sum_{i=1}^{M} \alpha V_i \);  

The adaptive value \( F(V) \) of \( V \) is calculated; \( F_{max} = F(V_{max}) = \max_{1 \leq i \leq N} F(V_i) \);  

if \( F(V)<F_{max} \) \( V_{max} = V \);  

\( t = t + 1; \}\}

The essence of this algorithm is a random search process based on subspace \( S \) determined by multiple parents. The difference between this algorithm and simple genetic algorithm is that the search space is determined by more points instead of two points. The adjustable parameters in this algorithm include \( N, M, a, b \), and its optimal setting depends on specific problems.

4.3 Calculation of model output
When the model structure, model parameters and model input are determined, the model expression can
be analyzed and the model output can be calculated. For the sake of convenient implementation, MSE is still used as the standard to measure whether the model sequence is close to the measured sequence in GP stage. We set \( n \) tree structures in the local bulletin board to record the node information of the algorithm tree of winners in the convergence process, which is called information tree. Usually, the generated formula can be formed by some simple operators through finite number of operations and compounding. Because modeling describes the problem of hierarchy and no fixed pattern, we use binary tree to describe the modeling operator.

Every evolutionary generation, the node information of the winner is recorded in these \( n \) information trees, and then a new generation of individuals is produced under the guidance of them. Therefore, non-winners can obtain the node information of winners according to a certain probability to complete the learning process. Two new trees are obtained by exchanging the corresponding subtrees with the hybridization point as the root node, and any new tree whose depth does not exceed \( D \) is selected as the hybrid offspring. Considering that the maximum Lyapunov index describes the exponential divergence rate of the adjacent tracks in the chaotic system, the longest prediction time \( L \) can be taken as the reciprocal of \( \lambda \), that is, \( L \approx \frac{1}{\lambda} \) can be taken as the upper time bound of the deterministic prediction of the system [12]. \( L \) can be appropriately larger when the data accuracy is relatively high.

### 4.4 Evaluation of model individuals

The evaluation strategy of model individual determines the evolution direction of GP. In GPM, the fitness value of individual \( i \) is determined by the following formula

\[
\text{fitness}(i) = \frac{1}{1 + \text{precision}(i)} \cdot \alpha(h)
\]

\( \text{fitness}(i) = \sum_{k=1}^{n} [x(k) - \hat{x}(k)] \)

Among them, \( x(k) \) represents the accuracy of the model represented by the individual, \( x(k) \) is the measured sequence, \( \hat{x}(k) \) is the output sequence of the model, and the inversion of change \( 1/(1 + \text{precision}(i)) \) makes the individual with smaller model error have higher fitness value.

At the same time, it can smooth the sharp fluctuation of model error and make GP run better. \( \alpha(h) \) is the adjustment factor, which is determined by the tree depth \( h \) and controls the complexity of the model, which makes GPM tend to look for those models with simple structure:

\[
\alpha(h) = \begin{cases} 
\text{abs}(h - h')/2^{h-2} & h \geq 4 \\
\text{abs}(h - h') \times 4 & h \leq 2 \\
\text{abs}(h - h') & \text{else}
\end{cases}
\]

\( h' \) indicates the expected tree depth, which is taken as 2.5 here.

### 5. Experimental analysis

In order to compare the effects of using different algorithms to build the ordinary differential equation system model of dynamic system, we take two actual time series as examples to build the American population growth model and the kinetic model of thermal decomposition reaction of chlorinated cyclohexane. For each example, the above algorithms were independently run for 25 times, and the average results were counted. All experiments were carried out on Pentium II microcomputer (266MHz).

GP algorithm and GA algorithm are used to run independently for 25 times, and the statistical
results shown in Table 1 are obtained, in which the complexity of the model is described by the number of nodes in the tree representation of differential equation.

| Algorithm | Average error | Minimum error | Maximum error | Average time /s |
|-----------|---------------|---------------|---------------|-----------------|
|           | Fitting error | Prediction error | Fitting error | Prediction error |                  |                  |                  |                  |
| GP        | 33.21045      | 32.05711      | 10.22821      | 2.37124         | 55.24711        | 54.63121        | 19              |
| GA        | 27.14250      | 29.30781      | 9.963277      | 3.37120         | 54.12729        | 53.67118        | 22              |

It can be seen from Table 1 that with the gradual improvement of the algorithm, the most obvious change is that the obtained model is getting more and more stable and better. On the one hand, the average fitting error and average prediction error of the model are obviously reduced, and on the other hand, the different model structures obtained from multiple runs are becoming less and less.

The fitting and prediction results of the model with the best parameter optimization are shown in Figure 1.

![Fig.1 Model fitting and prediction results](image)

It can be seen from Figure 1 that this model has good fitting and prediction results.

6. Conclusion
The research on evolution modeling of ordinary differential equations is still in the initial stage, and there are many unsolved problems, such as which ordinary differential equations numerical algorithm is the most suitable when calculating the fitness value of the model, and how to select their parameters (such as step size, starting header, etc.). The combination of GP technology and traditional search technology can enhance the performance of GP, but whether it can improve the performance of traditional search algorithm is not certain. After embedding the parameter optimization process of the model in GP, the performance has been significantly improved, which is far superior not only to the simple GP algorithm, but also to all traditional search algorithms and their combination classes. Of course, these conclusions are based on the specific problems we have solved, which can not be strictly proved in theory. Whether they are applicable to parallel GP researchers in solving other problems remains to be further verified. We hope that the research in this paper can provide some guidance for the algorithm design of these researchers.

References
[1] Sundnes J, Lines G T, Tveito A. Efficient solution of ordinary differential equations modeling electrical activity in cardiac cells. Mathematical Biosciences, vol. 172, no. 2, pp. 55-72, 2001.
[2] Haiyan Jiang, Zhao Kongnuan, Tang Liang, et al. Transactions of the Chinese Society of Agricultural Engineering no. Transactions of the CSAE), vol. 34, no. 21, pp. 176-184, 2018.
[3] Zhou S L, Liu Y Y. Co-evolutionary genetic optimization based ordinary differential equations identification for multi-input multi-output chaotic systems. IEEE, 2012:1745 - 1751.
[4] Mi Minghao, Qiu Nianhang, Lu Yuchao. Study on the optimum thickness of high temperature
protective clothing based on multi-objective programming. China Strategic Emerging Industries, vol. 000, no. 042, pp. 29-30, 2018.

[5] Yan Qing, Miao Zhuang, Xu Tiaobin, et al. Application of genetic algorithm based on Excel in mathematical modeling of college students. Journal of Shangrao Normal University, vol. 040, no. 003, pp. 18-24, 2020.

[6] Du hongqing, Chen Dewang, Huang yunhu, et al. fuzzy system optimization modeling method based on improved genetic algorithm and support degree. journal of intelligence science and technology, vol. 2, no. 02, pp. 82-88, 2020.

[7] Song Qiang. Modeling and optimization of improved hybrid genetic algorithm in MTVRPTW. Journal of Chongqing Jiaotong University no. Natural Science Edition), vol. 37, no. 09, pp. 82-89+137, 2018.

[8] Cao H, Kang L, Chen Y, et al. The Dynamic Evolutionary Modeling of HODEs for Time Series Prediction. Computers & Mathematics with Applications, vol. 46, no. 8-9, pp. 1397-1411, 2003.

[9] Liu Yanyun. Application of improved genetic algorithm in solving ordinary differential equations. Journal of Suzhou Institute of Education, vol. 21, no. 04, pp. 105-107, 2018.

[10] Sun Mingyue, Yuan Wenkai, Yu Zhongnan. Ecological Simulation of "Dragon" Species Based on Mathematical Modeling. China Strategic Emerging Industries, vol. 000, no. 002, pp. 98, 2020.

[11] Basarab-Horwath P, Zhdanov R Z. Initial-value problems for evolutionary partial differential equations and higher-order conditional symmetries. Journal of Mathematical Physics, vol. 42, no. 1, pp. 376-389, 2001.

[12] Mamontov E. Dynamic-equilibrium solutions of ordinary differential equations and their role in applied problems. Applied Mathematics Letters, vol. 21, no. 4, pp. 320-325, 2008.