IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 25, 2019, accepted April 14, 2019, date of publication May 2, 2019, date of current version May 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2913084

RNDEtree: Regulatory Network With Differential Equation Based on Flexible Neural Tree With Novel Criterion Function

BIN YANG[1] AND WENZHENG BAO[2]
[1]School of Information Science and Engineering, Zaozhuang University, Zaozhuang 277160, China
[2]School of Information and Electrical Engineering, Xuzhou University of Technology, Xuzhou 221018, China

Corresponding author: Wenzheng Bao (baowz55555@126.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61702445, in part by the Shandong Provincial Natural Science Foundation, China under Grant ZR2015PF007, in part by the Ph.D. Research Startup Foundation of Zaozhuang University under Grant 2014BS13, and in part by the Zaozhuang University Foundation under Grant 2015YY02.

ABSTRACT Gene regulatory network (GRN) could provide guidance for understanding the internal laws of biological phenomena and analyzing several diseases. Ordinary differential equation model, which owns continuity and flexibility, has been utilized to identify GRN over the past decade. In this paper, we propose a novel algorithm, which is named as RNDEtree, a nonlinear ordinary differential equation model based on a flexible neural tree to improve the accuracy of the GRN reconstruction. In this model, a flexible neural tree can be utilized to approximate the nonlinear regulation function of an ordinary differential equation model. Multiexpression programming is proposed to evolve the structure of a flexible neural tree, and the brainstorm optimization algorithm is utilized to optimize the parameters of the RNDEtree model. In order to improve the false-positive ratio of this method, a novel fitness function is proposed, in which sparse and minimum redundancy maximum relevance (mRMR) terms are considered when optimizing RNDEtree. The performances of our proposed algorithm can be evaluated by the benchmark datasets from the DREAM challenge and real biological dataset in *E. coli*. The experimental results demonstrate that the proposed method could infer more correctly GRN than the other state-the-art methods.

INDEX TERMS Gene regulatory network, flexible neural tree model, ordinary differential equation, mutual information, minimum redundancy maximum relevance.

I. INTRODUCTION

Research on gene regulatory network (GRN) could reveal the complex life phenomena from the viewpoint of system, and control the growth, heredity and variation of organism [1]–[3]. Due to rapid development in DNA microarray and next-generation sequencing technologies, the enormous gene expression data have provided the researchers lots of opportunities for gene regulatory network inference with computational and mathematical methods [4], [5].

Several methods have been devised to identify GRN, including Boolean network [6], Bayesian network [7], differential equation [8], Petri network [9], mutual information [10] and so on. Boolean model needs to discrete gene expression level, which could result in loss of information. Bayesian network (BN) model could not consider dynamic and feedback loop, which are the most important features of gene regulatory network. Dynamic Bayesian network (DBN) model solves this problem, but DBN need to spend a lot of time learning the conditional dependence relationships (regulatory relationships among genes). Mutual information (MI) is very simple, but it needs to assume that the samples between different time points are independent, and it could identify many indirect regulatory relationships and the accuracy is not high. Petri nets could not reflect the dynamic characteristics of networks. It is difficult to learn large-scale gene regulatory networks. The system of ordinary differential equation (ODE) belongs to a sophisticated and well established class of methods, which could capture the detailed GRN's dynamics due to its continuity and flexibility. The decoupled version of ODE could identify the regulatory relationships of each gene independently, which could greatly reduce the time of learning ODE model and facilitate large-scale biology network inference. Thus ODEs have

The associate editor coordinating the review of this manuscript and approving it for publication was Navanietha Krishnaraj Rathinam.

been mostly utilized to model biochemical networks [11]. The formal of ODE is divided into two classes. One is the linear ODE. Linear differential equation is simple, contains few parameters and needs few data. Wu et al. proposed a sparse additive linear ODE model to model gene expression data and infer GRN [12]. Gebert et al. proposed a system of piecewise linear differential equations to infer GRN and discrete approximation method of least squares was presented to evolve the parameters [13]. GRN, however is a complex system and has some characteristics, such as strong coupling, random, time-delayed, strongly nonlinear, etc. To accurately capture the properties of GRN, nonlinear ODE model were proposed, such as S-system model [14]–[17]. Mazur *et al.* reconstructed nonlinear differential equation model of gene regulation using stochastic sampling and Hill-type functions were added into the formal of ODE [18]. Dehghannasiri et al. proposed intrinsically Bayesian robust (IBR) Kalman filtering to optimize nonlinear ODE model for inferring a Yeast cell cycle network [19].

As the classical and typical approaches, neural network (NN) has been successfully and widely utilized to GRN inference with several decades, including recurrent neural networks (RNN) [20], recurrent Elman neural networks (RENN) [21] and neural fuzzy recurrent network (NFRN) [22]. Compared with the traditionally fully connected neural network, flexible neural tree (FNT) is more flexible and easier to approximate the unknown complex functions, and supports feature selection and over-layer connections [23]. The ODE model inferring GRN contains two parts: regulation function and the self-degradation part. In order to capture well the nonlinear of GRNs and integrate the advantages of ODE and NN, in this paper, FNT model is proposed to approximate the regulation function, namely ODE based on FNT (RNDEtree). Multi-expression programming (MEP) and brain storm optimization (BSO) algorithm are utilized to optimize the structure and parameters of RNDEtree model from expression data, respectively.

The characteristics and features of gene expression data contain two types. On the one hand, the error of the experimental equipment and the different operation processes of the researchers, the gene expression data contain noise. On the other hand, DNA microarray experiment can measure the expression levels of thousands of genes at the whole genome levels, and the scale of experiment is very small due to experiment cost and time. The number of genes is much larger than the number of experiments [24]. These factors limit the effective constructing gene regulatory networks. The minimum redundancy maximum relevance (mRMR) was proposed as a new feature extraction method to select important genes from microarray gene expression data [25]–[26]. In this paper, a novel criterion function based on sparse and mRMR terms is proposed to select the regulatory factors for each target gene when searching the optimal RNDEtree model. As a filtering method, sparse and mRMR terms could reduce sharply the number of candidate regulatory factors for each target gene.

Artificial gene expression data from the DREAM3 challenge about *Yeast* and *E.coli* knock-out genes with size 50 and 100 and one real gene expression dataset about *E. coli* downloaded from RegulonDB are utilized to test the performance of our proposed algorithm. Results reveal that our algorithm has the ability of identifying gene regulatory network correctly.

II. METHOD

A. RNDEtree MODEL

The ordinary differential equation model is a common dynamic system and usually utilized to simulate the evolution of biological macromolecules with time. In order to identify gene regulatory network, an ODE can be utilized to represent the regulatory relationships between each target gene and its regulatory factors. The formal of one ODE is described as follows:

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \ldots x_n) - \beta_i x_i. \tag{1}$$

where x_i is the express level of gene i, β_i is the self-degradation rate. $f_i(\cdot)$ means the regulation function containing linear, piecewise linear, pseudo linear (Sigmoid) and nonlinear functions. The number of parameters and topology in $f_i(\cdot)$ determine the regulation strengths. To better model regulation function, flexible neural tree model is proposed to model the regulation function $f_i(\cdot)$. The formal of ODE based on FNT (RNDEtree) is described as Eq.(2) and in Fig. 1.

$$\frac{dx_i}{dt} = FNT_i - \beta_i x_i \tag{2}$$

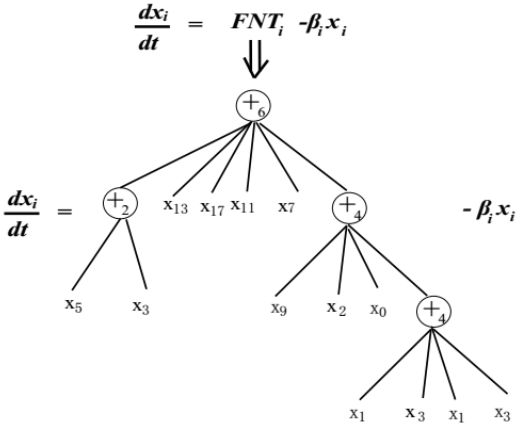

FIGURE 1. The formal of differential equation model based on flexible neural tree.

B. RNDEtree MODEL OPTIMIZATION

1) THE STRUCTURE OF FNT MODEL

The FNT model is a novel and flexible multi-layer feedforward NN proposed by Chen in year 2005, which could select feature automatically and connect by over-layer style [27]. In this model, the input variables, the number of layers, the structure of each layer and output variables could be created

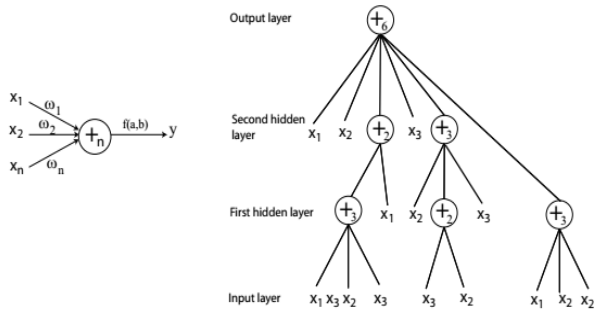

FIGURE 2. A flexible neuron operator (left) and an example of FNT model (right).

randomly with function symbol set F and terminal symbol set T, which are described as follows:

$$\begin{cases} F = \{+_2, +_3, \ldots +_N\}, \\ T = \{x_1, x_2, \ldots, x_n\}. \end{cases} \tag{3}$$

where $+_i$ $(i = 2, 3, \ldots, N)$ can be treated as a function symbol and represents the addition of i input numbers. $x_i (i = 1, 2, \ldots, n)$ is a terminal symbol. An example of FNT model is depicted in Fig. 2. The function node as a flexible neuron model (see Fig. 2) could be calculated as follows.

$$e_i = \sum_{j=1}^{i} w_i \times x_j. \tag{4}$$

$$o_i = f(a_i, b_i, e_i) = e^{-(\frac{e_i - a_i}{b_i})^2}. \tag{5}$$

where x_j is the input variable, w_j is the weight and $f(\cdot)$ is the activation function.

2) STRUCTURE OPTIMIZATION OF FNT MODEL

Multi-expression programming (MEP) is a structure-based swarm evolutionary algorithm proposed by Oltean. Compared to genetic programming (GP), each chromosome of MEP is linearly encoded, contains multiple solutions and achieves code reuse [28]. So MEP has been widely applied in many areas, such as image processing [29], bioinformatics [30], and time series prediction [31].

In MEP, each chromosome includes multiple genes. The length of chromosome is equal to the number of genes. Each gene contains three parts: gene label, function symbol or terminal symbol and gene pointers of operands. Chromosome could be created randomly according to the predefined function symbol set F and terminal symbol set T. The symbol of the first gene in a chromosome must be terminal symbol. The symbols of the other genes could be selected randomly from F and T. If function symbol is selected, the pointers of the function operands are created randomly. This paper utilizes MEP to search the optimal FNT structure of RNDEtree model. Suppose that function symbol set is $F = \{+2, +3, +4\}$, terminal symbol set is $T = \{x_1, x_2, x_3, x_4, x_5\}$, and the number of genes is set as 8. FNT structures could be encoded as the chromosome of MEP, which is represented in Fig. 3. Each gene could be decoded into a FNT model,

1:	x1	
2:	x3	
3:	x5	
4:	+2	1 3
5:	x2	
6:	+3	5 2 4
7:	x4	
8:	+4	5 7 6 1

FIGURE 3. An example of the chromosome for representing FNT structures in MEP.

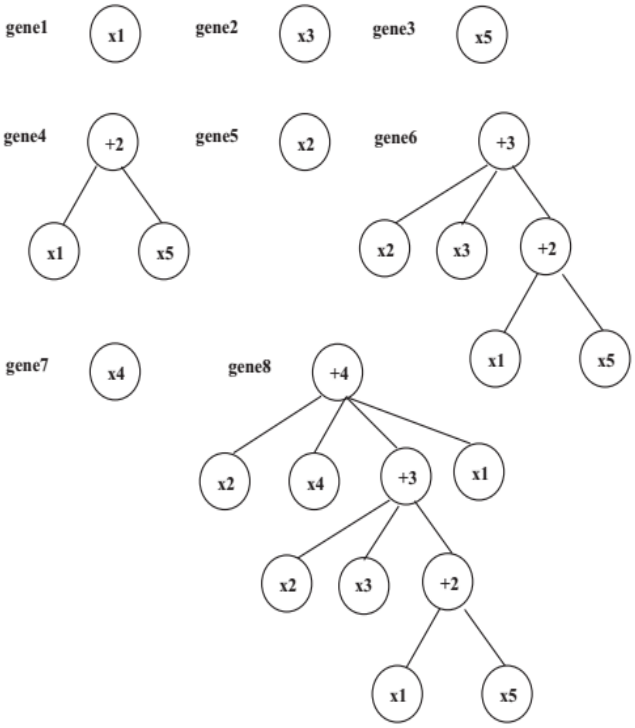

FIGURE 4. The tree structures of all genes of the chromosome in MEP.

which is described in Fig. 4. Each gene is a candidate FNT model. The fitness values of all genes in a chromosome are calculated, and the best fitness value is selected as the fitness value of the chromosome.

In order to search the best structure of FNT, the following genetic operators are utilized [31].

(1) Selection. Binary tournament selection is utilized to select the better chromosomes to the next generation.

(2) Crossover. According to crossover probability p_c, two parent chromosomes are chosen. The partial genes of two parents are exchanged in order to create two offsprings.
(3) Mutation. According to mutation probability p_m, the symbol of each gene of parent chromosome may be changed by another symbol selected randomly. After mutation, the symbol of the first gene must be a terminal symbol.

3) PARAMETER OPTIMIZATION OF RNDEtree MODEL

Brain storm optimization (BSO) is utilized to search the best parameter set of RNDEtree model, containing weights (w_i), activation function parameters (a_i, b_i) and self-degradation rate (β_i). BSO is a novel swarm intelligence algorithm based on human brain storming process, which was presented by Shi in year 2011 [32]. In this algorithm, the population is divided into K classes and the individuals in each class are optimized. By mutation operation, local search is implemented in order to obtain the local optimal solution of each class [33]. The global optimum solution is searched through inter-class collaboration. The process of BSO optimizing the parameters of RNDEtree model is depicted in **Algorithm 1**.

C. FITNESS FUNCTION

In order to search the optimal RNDEtree model by evolutionary method, mean square error (MSE) is employed as fitness function in this work.

$$F_i = \frac{1}{T}\sum_{t=1}^{T}(z_{it} - z_{it}^{'})^2. \tag{6}$$

where z_{it} and $z_{it}^{'}$ are real and forecasted expression data of $t-th$ gene at $t-th$ sample point, respectively.

Due to that GRN structure has the characteristics of sparsity and small-world network, the number of regulatory factors of each target gene is very small, which is very less than the number of the candidate regulatory factors, so this paper adds two filtering terms into MSE in order to reduce the number of candidate regulators.

(1) L_1 regularizer

With the L_1 regularization term, the criterion function is described as followed [34]–[35].

$$F_i = \frac{1}{T}\sum_{t=1}^{T}(z_{it} - z_{it}^{'})^2 + \alpha \left\| W_i \right\|. \tag{7}$$

where $\alpha \left\| W_i \right\|$ is the sparse term, $\left\| W_i \right\|$ is L_1 regularizer of weight parameter vector W_i from $i-th$ RNDEtree model, and α is a sparse coefficient.

(2) Minimum redundancy maximum relevance (mRMR)

Generally mutual information (MI) could be utilized to measure the regulatory relationship between two genes in GRN, so this paper utilizes MI to compute relevancy and redundancy among genes in order to select the regulators with the maximum relevant and the minimum redundant for each target gene.

Algorithm 1 Pseudo Code of BSO Optimizing the Parameters of RNDEtree Model

```
Define probability p1,
Count the number of parameters in RNDEtree model n;
Initialize N individuals [X1, X2, ..., XN] (Xi = (xi^1, xi^2, ..., xi^n)) with the n dimension;
t = 0;
while i < tmax do
  N individuals are divided into K classes;
  for i = 1; i ≤ K; i++ do
    Calculate the fitness values of individuals in the i-th class;
    Sort the individuals in the i-th class according to fitness values;
    Central individual center_i ← select the optimal individual in the i-th class;
  end for
  r=rand(0,l);
  if r < p1 then
    k ← rand(1,K); // Select a class randomly;
    center_k ← center_k + Guassian(0, 1);
  end if
  for i = 1; i ≤ N; i++ do
    r-rand(1,4); // Select a kind of mutation method randomly;
    c1 ← rand(1,K); // Select a class randomly;
    c2 ← rand(1,K);
    if r == 1 then
      X_new ← center_c1;
    else if r == 2 then
      X_new ← select a individual from the C1 class;
    else if r == 3 then
      λ = rand(0, 1); X_new ← λ × center_c1 + (1 − λ)× center_c2;
    else
      a ← select a individual from the c1 class;
      b ← select a individual from the c2 class;
      λ = rand(0, 1); X_new ← λ × a + (1 − λ) × b;
    end if
    ε ← log Sigmoid ((t_max − t)/(2k));// k is the gradient X_new ← X_new + ε × Guassian(0, 1);
    Compare the fitness values of Xi and X_new and retain the best individual.
  end for
end while
```

Relevant term Rl is described as following [25], [36]:

$$Rl_i = \frac{1}{m}\sum\nolimits_{j\in\Omega_i} I(X_i, X_j). \tag{8}$$

where Ω_i represents the candidate regulatory factor set of $i-th$ target gene, which includes m regulators. $I(X_i, X_j)$ is MI value between gene i and gene j, which could be

calculate as follows.

$$I(X, Y) = p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \tag{9}$$

where $p(x)$ and $p(y)$ are the marginal probabilities of two genes X and Y, respectively, $p(x, y)$ is the joint probability of gene X and gene Y.

Gaussian kernel probability density estimator is utilized to estimate the marginal and joint probability, and MI can be given as followed [10].

$$I(X, Y) = \frac{1}{2} \log(\frac{\sigma_X^2 \sigma_Y^2}{|C(X, Y)|}). \tag{10}$$

where $|C(X, Y)|$ is the determinant of the covariance matrix. High MI value means that regulatory factor regulates target gene with a large probability.

Redundant term Rd is described as following [25], [36]:

$$Rd_i = \frac{1}{m^2} \sum\nolimits_{k,j \in \Omega_i} I(X_k, X_j). \tag{11}$$

With the mRMR term, the criterion function is described as followed.

$$F_i = \frac{\frac{1}{T} \sum_{t=1}^{T} (z_{it} - z'_{it})^2}{1 + Rl_i - Rd_i} + \alpha \left\| W_i \right\|. \tag{12}$$

where Rl_i and Rd_i are relevant and redundant terms, respectively.

D. FLOWCHART OF RNDEtree FOR GENE REGULATORY NETWORK INFERENCE

(1) Suppose that gene expression data $[D_1, D_2, \ldots, D_m]$ contains m genes and each gene contains n time points ($D_i = [D_i^1, D_i^2, \ldots, D_i^n]$). The regulatory relationships of each gene could be inferred by RNDEtree model independently. At first gene number i is set as 1, which means that regulatory relationships of the first gene will be inferred.

(2) Create the learning sample dataset. Gene expression data D_i of gene i is set as output data, while gene expression data $[D_1, D_2, \ldots, D_{i-1}, D_{i+1}, \ldots, D_m]$ of other genes are set as input vector. With the learning sample dataset, a hybrid evolutionary algorithm is utilized to optimize RNDEtree model, which is described as follows.

1) Initialize the RNDEtree population containing FNT structures and the corresponding parameters.
2) The fitness values of RNDEtree population are calculated by Eq.(12). If the optimal RNDEtree model is achieved, go to Step (3).
3) MEP is used to evolve the FNT structure in RNDEtree model. At some generations, BSO is used to evolve the parameters of RNDEtree model. Go to step 2).

(3) According to the optimal RNDEtree model, regulatory relationships of gene i are identified. If gene j is contained in RNDEtree model, gene i is regulated by gene j.

(4) $i = i + 1$. If $i \leq m$, goto step (2); otherwise goto (5).

(5) The regulations of all genes are integrated in order to obtain overall GRN.

III. EXPERIMENTS

A. DATA AND CRITERIONS DESCRIPTION

The proposed method is applied to four artificial datasets from the DREAM3 challenge about *Yeast* and *E.coli* knockout genes with size 50 and 100 [37] and one real gene expression dataset from *E. coli* [38]. *E.coli* and *Yeast* datasets in DREAM3 were supplied for accessing the advantage and shortcoming of network identification algorithms. These databases have been widely utilized as the benchmark data sets of the GRN reconstruction methods assessment. In DREAM3, *Yeast* and *E.coli* gene expression datasets with network size 50 and 100 contain sample number 50 and 100, respectively. The real GRN is from RegulonDB [38] (version 8.2), and includes 3306 regulatory relationships among 177 regulatory genes and 1532 target genes, which have been verified by biochemistry experiments. The expression data used in this experiment is downloaded from the Many Microarrays M3D database [39] (Microbe), and the version is version 4 build 6, which includes 907 experiments and 4297 genes.

		True GRN: Positive	True GRN: Negative
Inferred GRN	Positive	TP	FP
	Negative	FN	TN

FIGURE 5. TP, TN, FP, and FN.

In order to evaluate the performance of our proposed, five criterions (true positive rate (TPR), false positive rate (FPR), positive predictive value (PPV), Accuracy (ACC) and F-score) are utilized, which are defined as followed.

$$\begin{aligned} TPR &= \frac{TP}{TP + FN}, \\ FPR &= \frac{FP}{FP + TN}, \\ PPV &= \frac{TP}{TP + FP}, \\ ACC &= \frac{TP + TN}{TP + FP + TN + FN}, \\ F - score &= 2PPV \times \frac{TPR}{PPV + TPR}. \end{aligned} \tag{13}$$

where true positive (TP), true negative (TN), false positive (FP) and false negative (FN) are calculated according to Fig. 5 [40]. Regulatory relationships in GRN are marked

as positive samples, while non-regulatory relationships are marked as negative samples.

B. PARAMETER SELECTED

We first test different values of sparse coefficient α and evaluate its effect on TPR, FPR, PPV, ACC and F-score. In order to select proper value, the values of α are ranged from [0, 0.05] at an interval of 0.005. The testing data are from DREAM3 datasets about *Yeast* and *E.coli* knock-out genes with size 50 and 100.

TABLE 1. Performance of our method with different values of α on *Yeast* network with size 50 in DREAM3.

α	TPR	FPR	PPV	ACC	F-score
0	0.63636	0.10957	0.15858	0.86245	0.25389
0.005	0.57143	0.077118	0.19383	0.89388	0.28947
0.01	0.48052	0.062368	0.2	0.90816	0.28244
0.015	0.48052	0.049305	0.24026	0.92082	0.32035
0.02	0.48052	0.046776	0.25	0.92327	0.32889
0.025	0.44156	0.041719	0.25564	0.92816	0.32381
0.03	0.38961	0.040877	0.23622	0.92898	0.29412
0.035	0.4026	0.037927	0.2562	0.93184	0.31313
0.04	0.37662	0.034556	0.26126	0.9351	0.30851
0.045	0.36364	0.045933	0.20438	0.920408	0.26168
0.05	0.33766	0.038768	0.22034	0.93102	0.26667

TABLE 2. Performance of our method with different values of α on *E.coLi* network with size 50 in DREAM3.

α	TPR	FPR	PPV	ACC	F-score
0	0.66129	0.10637	0.13898	0.87102	0.22969
0.005	0.64516	0.064489	0.20619	0.91184	0.3125
0.01	0.56452	0.053183	0.22605	0.92286	0.3228
0.015	0.58065	0.044389	0.25352	0.93143	0.35294
0.02	0.5	0.04062	0.24219	0.9351	0.32632
0.025	0.58065	0.036851	0.29032	0.93878	0.3871
0.03	0.45161	0.034757	0.25225	0.94082	0.3237
0.035	0.43548	0.03392	0.25	0.94163	0.31765
0.04	0.51948	0.023639	0.19048	0.89918	0.27875
0.045	0.54545	0.022061	0.19718	0.89878	0.28966
0.05	0.46753	0.021061	0.17391	0.89878	0.25352

The results are listed in Table 1, Table 2, Table 3 and Table 4. From the results, the conclusions are described as followed.

(1) Sparse coefficient α plays an important role in the performance of our proposed inferred algorithm. With the increase of the sparse coefficient, the algorithm selects fewer regulators, the ratio of true positive sides (TPR) is less, and the ratio of the false positive sides (FPR) is also getting smaller.

(2) By observing the results of F-score, it can be found that the method performs better when the sparse coefficients are selected from the range [0.005, 0.025]. Compared with

TABLE 3. Performance of our method with different values of α on *Yeast* network with size 100 in DREAM3.

α	TPR	FPR	PPV	ACC	F-score
0	0.59036	0.052394	0.16118	0.93172	0.25323
0.005	0.48796	0.031642	0.20823	0.95212	0.29189
0.014	0.44578	0.024142	0.23948	0.95949	0.31158
0.015	0.40361	0.021368	0.24364	0.96222	0.30386
0.02	0.38554	0.019519	0.25197	0.96404	0.30476
0.025	0.36145	0.017875	0.25641	0.96566	0.3
0.03	0.34337	0.017773	0.24783	0.96576	0.28788
0.035	0.33735	0.016026	0.26415	0.96747	0.2963
0.04	0.29518	0.014999	0.25128	0.96848	0.27147
0.045	0.3313	0.015513	0.26699	0.96798	0.2957
0.05	0.31928	0.014794	0.26904	0.96869	0.29201

TABLE 4. Performance of our method with different values of α on *E.coLi* network with size 100 in DREAM3.

α	TPR	FPR	PPV	ACC	F-score
0	0.664	0.059028	0.12576	0.92909	0.21146
0.005	0.56	0.032737	0.17949	0.95505	0.27184
0.015	0.52	0.023325	0.22184	0.96434	0.311
0.02	0.42169	0.025478	0.22013	0.95818	0.28926
0.025	0.38554	0.021985	0.23022	0.96162	0.28829
0.03	0.38554	0.019519	0.25197	0.96404	0.30476
0.035	0.35542	0.018184	0.25	0.96535	0.29535
0.04	0.33735	0.016129	0.26129	0.96737	0.29551
0.045	0.34337	0.014896	0.28218	0.96859	0.30978
0.05	0.248	0.011049	0.22302	0.97646	0.23485

the results of the two kinds of networks, it can be seen that, the number of genes has a little effect on performance.

Based on the above analysis, we choose the value of α as 0.015 for our following inference works, since it is the closest number to the middle of the optimal range.

C. PERFORMANCE RESULTS

To test the validation of our method, LASSO [41], random forest (GENIE3) with parameters 'sqrt' [42], ARACNE [43] and ODE [44] are also utilized to infer gene regulatory network with the same data.

1) SIMULATED DATA

Yeast and *E.coli* gene expression data with 50 and 100 genes in DREAM3 are utilized to evaluate our method. The inferred results by LASSO, GENIE3, ARACNE, ODE and RNDEtree are depicted in Fig. 6, Fig. 7, Fig. 8 and Fig. 9, respectively. From Fig. 6, it can be seen that ARACNE algorithm has the highest TPR, and our method has the smallest FPR, the highest PPV, ACC and F-score for *Yeast* network with 50 genes inference. Fig. 7 shows that in terms of TPR, FPR, PPV, ACC and F-score, our proposed method has the best performance among these five methods for *Yeast* network

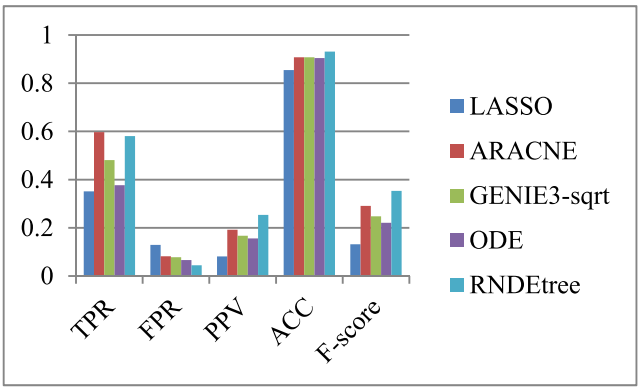

FIGURE 6. Comparison of different methods on *Yeast* network with size 50 in DREAM3.

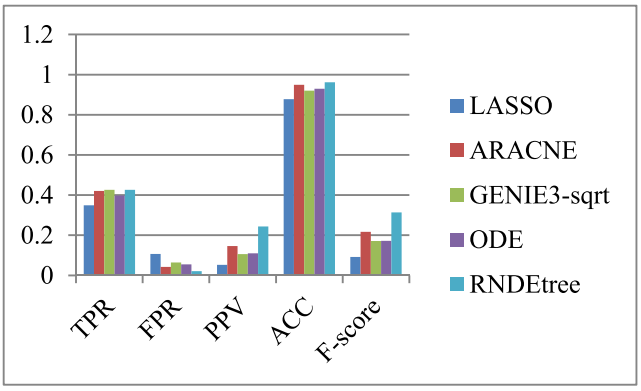

FIGURE 7. Comparison of different methods on *Yeast* network with size 100 in DREAM3.

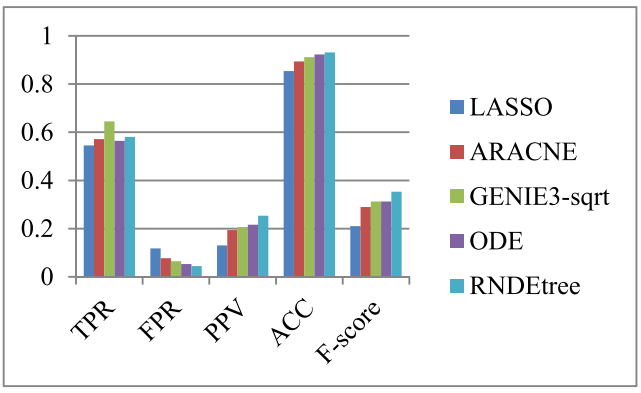

FIGURE 8. Comparison of different methods on *E.coli* network with size 50 in DREAM3.

with 100 genes inference. For the construction of *E.coli* networks with 50 and 100 genes (Fig. 8 and Fig. 9), in terms of FPR, PPV, ACC and F-score, our method performs best. The TPR is second to GENIE3 method. In sum, most results of our proposed method are superior to other comparison algorithms.

2) REAL GENE EXPRESSION DATA

In this section, a sub network is extracted from real *E. coli* network, which consists of 114 target genes, 127 regulatory factors and 227 regulatory relationships. Each target

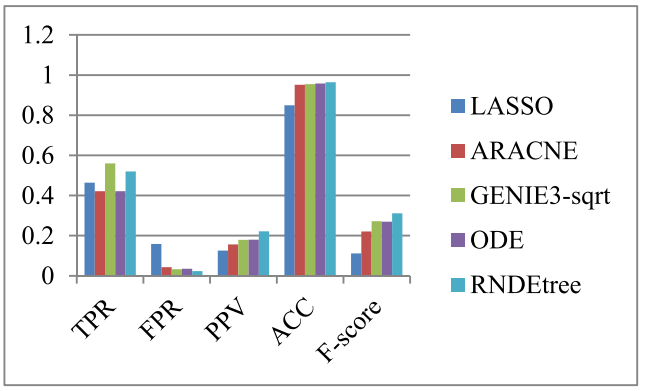

FIGURE 9. Comparison of different methods on *E.coli* network with size 100 in DREAM3.

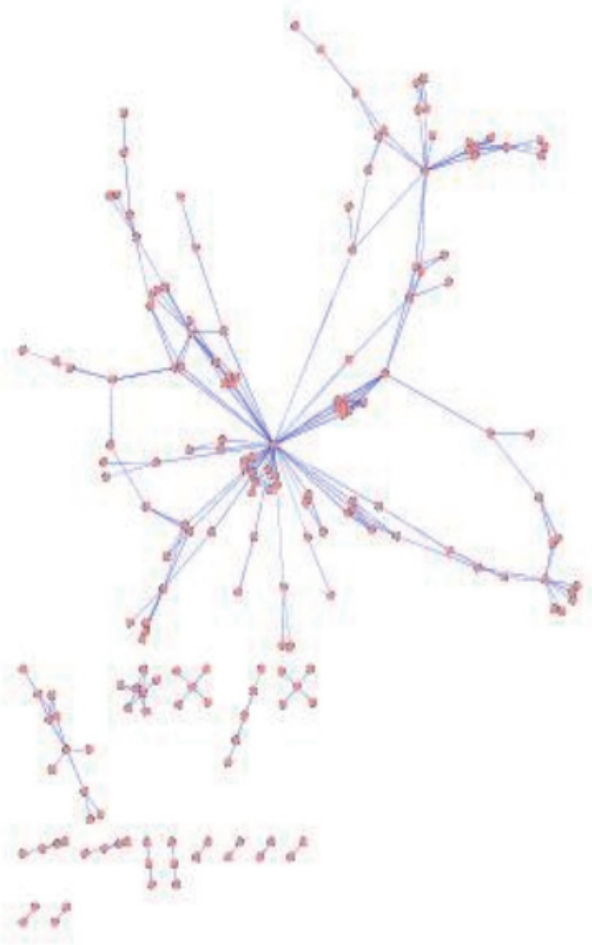

FIGURE 10. The sub network structure.

gene contains two regulatory factors on average, which is consistent with the characteristics of the main network. The network structure is depicted in Fig. 10. Through several runs, Table 5 lists the averaged performance of the networks using five methods, which reveals that in terms of TPR, FPR, PPV, ACC and F-score, our proposed method are all the best among these five methods, which are 0.49123, 0.016632, 0.28283, 0.976903 and 0.35897, respectively. Although our method

TABLE 5. Comparisons of different methods on networks with real gene expression data.

	TPR	FPR	PPV	ACC	F-score
RNDEtree	0.49123	0.016632	0.28283	0.976903	0.35897
GENIE3-sqrt [42]	0.4386	0.01892	0.23256	0.96842	0.30395
ARACNE [43]	0.49123	0.020525	0.2383	0.96684	0.32092
ODE [44]	0.3524	0.02787	0.13821	0.96638	0.19855
LASSO [41]	0.30702	0.03096	0.11475	0.95654	0.16706

has the same TPR as ARACNE, FPR is 19 % smaller, PPV is 18.7 % higher, ACC is 1.04 % higher and F-score is 11.9 % higher than those of ARACNE algorithm, respectively.

From the results of GRN inferred with simulation and real expression datasets, it could be clearly seen that LASSO has the lowest TPR, while ARACNE, GENIE3 and RNDEtree have higher TPR. Because LASSO is based on the idea of linear regression and cannot simulate well complex regulatory relationships, there are fewer real regulatory relationships identified. ARACNE uses threshold to select the regulatory relationship with a high confidence. If the threshold is low, more real regulatory relationships can be identified. GENIE3 could get more real regulatory relationships due to the idea of problem decomposition and ensemble. RNDEtree utilizes the nonlinear differential equation model to simulate the complex regulatory relationships among genes, and utilizes mRMR term to select regulatory factors with high reliability, so TPR is also relatively high. Compared with ARACNE and GENIE3, RNDEtree adds sparse term, which could delete some redundant false-positive regulatory relationships. So RNDEtree has the smallest FPR among the five methods, which is to mean that the identified regulatory networks contain the least false positive regulatory relationships. Overall, RNDEtree has the largest F-score, so it could identify more accurate gene regulatory networks.

Although ODE and RNDEtree are based on nonlinear differential equation model, a novel criterion function (Eq. 15) is utilized in RNDEtree. From the performances of ODE and RNDEtree, it could be shown that RNDEtree performs better in terms of TPR, FPR, PPV, ACC and F-score, which reveals that our proposed criterion function plays an important role in improving the accuracy of GRN inferred.

D. EFFECT OF SPARSE TERM

Table 1, Table 2, Table 3 and Table 4 show the effect of sparse term on RNDEtree. When α is set as 0, it means that RNDEtree has no sparse term. We chose the value of α as 0.015 to compare with it. By comparison, we can find that RNDEtree without sparse term can identify more real regulatory relationships. After adding sparse term, FPR decreases, but TPR also decreases, which indicate that sparse term could delete some true-positive regulatory relationships while deleting false-positive regulatory relationships. As a compromise criterion between TPR and FPR, F-score could be improved 20%-60% after adding sparse term. Although sparse term could delete some true-positive regulatory relationships, more false-positive relationships are deleted to make the network structure more accurate. It is necessary to add sparse term.

IV. CONCLUSIONS

In this work, flexible neural tree instead of the nonlinear regulation function of ordinary differential equation model for gene regulatory network is built from gene expression data. Multi expression programming and brain swarm optimization algorithm are used to optimize the FNT structure and ODE parameters, respectively. Moreover, a new fitness function based on sparse and minimum redundancy maximum relevance terms is proposed to improve the accuracy of GRN. Experimental results reveal that our proposed method is better than other state-of-the-art methods (LASSO, ARACNE and GENIE3). From the results, it can be also seen that our GRN inferred by our method has convincing TPR and the smallest FPR. This is because that our proposed fitness function could reduce the candidate regulatory set and FNT could model well complex regulatory relationships and select automatically the proper regulatory factors with gene expression data. In future work, parallel technology will be introduced to speed up the evolutionary process of RNDEtree model.

REFERENCES

[1] B.-S. Chen, S.-K. Yang, C.-Y. Lan, and Y.-J. Chuang, "A systems biology approach to construct the gene regulatory network of systemic inflammation via microarray and databases mining," *BMC Med Genomics*, vol. 1, p. 46, Sep. 2008.

[2] G. Chen, M. J. Cairelli, H. Kilicoglu, D. Shin, and T. C. Rindflesch, "Augmenting microarray data with literature-based knowledge to enhance gene regulatory network inference," *PLoS Comput. Biol.*, vol. 10, no. 6, 2014, Art. no. e1003666.

[3] Z. P. Liu, "Reverse engineering of genome-wide gene regulatory networks from gene expression data," *Current Genomics*, vol. 16, no. 1, pp. 3–22, 2015.

[4] J. N. Bazil *et al.*, "The inferred cardiogenic gene regulatory network in the mammalian heart," *PLoS ONE*, vol. 9, no. 6, 2014, Art. no. e100842.

[5] F. Emmert-Streib, M. Dehmer, and B. Haibe-Kains, "Gene regulatory networks and their applications: Understanding biological and medical problems in terms of networks," *Frontiers Cell Develop. Biol.*, vol. 2, p. 38, Aug. 2014.

[6] P. Trairatphisan, A. Mizera, J. Pang, A. A. Tantar, J. Schneider, and T. Sauter, "Recent development and biomedical applications of probabilistic Boolean networks," *Cell Commun. Signaling*, vol. 11, p. 46, Jul. 2013.

[7] E. Acerbi, T. Zelante, V. Narang, and F. Stella, "Gene network inference using continuous time Bayesian networks: A comparative study and application to Th17 cell differentiation," *BMC Bioinf.*, vol. 15, no. 1, p. 387, 2014.

[8] Y. K. Wang, D. G. Hurley, S. Schnel, C. G. Print, and E. J. Crampin, "Integration of steady-state and temporal gene expression data for the inference of gene regulatory networks," *PLoS ONE*, vol. 8, no. 8, 2013, Art. no. e72103.

[9] M. Durzinsky, A. Wagler, and W. Marwan, "Reconstruction of extended Petri nets from time series data and its application to signal transduction and to gene regulatory networks," *BMC Syst. Biol.*, vol. 5, p. 113, Jul. 2011.

[10] X. Zhang, J. Zhao, J.-K. Hao, X.-M. Zhao, and L. Chen, "Conditional mutual inclusive information enables accurate quantification of associations in gene regulatory networks," *Nucleic Acids Res.*, vol. 43, no. 5, p. e31, 2015.

[11] B. Yang, M. Jiang, and Y. Chen, "A novel hybrid framework for reconstructing gene regulatory networks," *Int. J. Hybrid Inf. Technol.*, vol. 6, no. 5, pp. 255–268, 2013.

[12] H. Wu, T. Lu, H. Xue, and H. Liang, "Sparse additive ordinary differential equations for dynamic gene regulatory network modeling," *J. Amer. Stat. Assoc.*, vol. 109, no. 506, pp. 700–716, 2014.
[13] J. Gebert, N. Radde, and G.-W. Weber, "Modeling gene regulatory networks with piecewise linear differential equations," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1148–1165, 2007.
[14] Y.-T. Hsiao and W.-P. Lee, "Reverse engineering gene regulatory networks: Coupling an optimization algorithm with a parameter identification technique," *BMC Bioinf.*, vol. 15, p. S8, Dec. 2014.
[15] N. Noman and H. Iba, "Inference of gene regulatory networks using s-system and differential evolution," in *Proc. Conf. Genetic Evol. Comput.*, Washington DC, USA, 2005, pp. 439–446.
[16] E. O. Voit and J. Almeida, "Decoupling dynamical systems for pathway identification from metabolic profiles," *Bioinformatics*, vol. 20, no. 11, pp. 1670–1681, 2004.
[17] B. Yang, W. Zhang, H. Wang, C. Song, and Y. Chen, "TDSDMI: Inference of time-delayed gene regulatory network using S-system model with delayed mutual information," *Comput. Biol. Med.*, vol. 72, pp. 218–225, May 2016.
[18] J. Mazur, D. Ritter, G. Reinelt, and L. Kaderali, "Reconstructing nonlinear dynamic models of gene regulation using stochastic sampling," *BMC Bioinf.*, vol. 10, p. 448, Dec. 2009.
[19] R. Dehghannasiri, M. S. Esfahani, and E. R. Dougherty, "Inference of nonlinear ODE-based gene regulatory networks via intrinsically Bayesian robust Kalman filtering," in *Proc. 7th ACM Int. Conf. Bioinf., Comput. Biol., Health Inform.*, Seattle, WA, USA, 2016, pp. 542–543.
[20] R. Xu, D. L. Wunsch, II, and R. L. Frank, "Inference of genetic regulatory networks with recurrent neural network models using particle swarm optimization," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 4, no. 4, pp. 681–692, Oct./Dec. 2007.
[21] S. I. Ao and V. Palade, "Ensemble of Elman neural networks and support vector machines for reverse engineering of gene regulatory networks," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1718–1726, 2011.
[22] I. A. Maraziotis, A. Dragomir, and A. Bezerianos, "Gene networks reconstruction and time-series prediction from microarray data using recurrent neural fuzzy networks," *IET Syst. Biol.*, vol. 1, no. 1, pp. 41–50, Jan. 2007.
[23] B. Yang, Y. Chen, and M. Jiang, "Reverse engineering of gene regulatory networks using flexible neural tree models," *Neurocomputing*, vol. 99, pp. 458–466, Jan. 2013.
[24] L. Z. Liu, F. X. Wu, and W. J. Zhang, "Reverse engineering of gene regulatory networks from biological data," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 2, no. 5, pp. 365–385, 2012.
[25] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *J. Bio. Comput. Biology*, vol. 3, no. 2, pp. 185–205, 2005.
[26] Q. Zou, J. Zeng, L. Cao, and R. Ji, "A novel features ranking metric with application to scalable visual and bioinformatics data classification," *Neurocomputing*, vol. 173, pp. 346–354, Jan. 2016.
[27] Y. Chen, B. Yang, J. Dong, and A. Abraham, "Time-series forecasting using flexible neural tree model," *Inf. Sci.*, vol. 174, nos. 3–4, pp. 219–235, 2005.
[28] M. Oltean and C. Grosan, "Evolving evolutionary algorithms using multi expression programming," in *Proc. 7th Eur. Conf. Artif. Life*, Dortmund, Germany, 2003, pp. 651–658.
[29] W. Wang, W. Lin, and Q. Li, "Image retrieval based on multi expression programming algorithms," *Crit. Criminol.*, vol. 21, no. 2, pp. 157–176, 2013.
[30] A. Baykasoğlu and L. Özbakir, "MEPAR-miner: Multi-expression programming for classification rule mining," *Eur. J. Oper. Res.*, vol. 183, no. 2, pp. 767–784, 2007.
[31] C. Grosan, A. Abraham, V. Ramos, and S. Y. Han, "Stock market prediction using multi expression programming," in *Proc. Portuguese Conf. Artif. Intell.*, Covilha, Portugal, 2007, pp. 73–78.
[32] Y. Shi, "Brain storm optimization algorithm," *Artif. Intell. Rev.*, vol. 6728, no. 3, pp. 1–14, 2011.
[33] S. Cheng, Q. Qin, J. Chen, and Y. Shi, "Brain storm optimization algorithm: A review," *Artif. Intell. Rev.*, vol. 46, no. 4, pp. 445–458, 2016.
[34] T. Hasegawa, R. Yamaguchi, M. Nagasaki, S. Miyano, and S. Imoto, "Inference of gene regulatory networks incorporating multi-source biological knowledge via a state space model with l1 regularization," *PLoS ONE*, vol. 9, no. 8, 2014, Art. no. e105942.
[35] K. Kojima, A. Fujita, T. Shimamura, S. Imoto, and S. Miyano, "Estimation of nonlinear gene regulatory networks via L1 regularized NVAR from time series gene expression data," *Genome Inf.*, vol. 20, pp. 37–51, Feb. 2008.
[36] H. M. Alshamlan, G. H. Badr, and Y. A. Alohali, "Genetic Bee Colony (GBC) algorithm: A new gene selection method for microarray cancer classification," *Comput. Biol. Chem.*, vol. 56, pp. 49–60, Jun. 2015.
[37] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *Proc. Nat. Acad. Sci. USA*, vol. 107, no. 14, pp. 6286–6291, 2010.
[38] S. Gama-Castro *et al.*, "RegulonDB version 7.0: Transcriptional regulation of Escherichia coli K-12 integrated within genetic sensory response units (Gensor Units)," *Nucleic Acids Res.*, vol. 39, pp. D98–D105, Jan. 2011.
[39] J. J. Faith *et al.*, "Many microbe microarrays database: Uniformly normalized Affymetrix compendia with structured experimental metadata," *Nucleic Acids Res.*, vol. 36, pp. D866–D870, Jan. 2008.
[40] B. Yang, Y. Chen, W. Zhang, J. Lv, W. Bao, and D.-S. Huang, "HSCVFNT: Inference of time-delayed gene regulatory network based on complex-valued flexible neural tree model," *Int. J. Mol. Sci.*, vol. 19, no. 10, p. 3178, 2018.
[41] G. Geeven, R. E. van Kesteren, A. B. Smit, and M. C. de Gunst, "Identification of context-specific gene regulatory networks with GEMULA—Gene expression modeling using LAsso," *Bioinformatics*, vol. 28, no. 2, pp. 214–221, 2012.
[42] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, "Inferring regulatory networks from expression data using tree-based methods," *PLoS ONE*, vol. 5, Sep. 2010, Art. no. e12776.
[43] A. A. Margolin *et al.*, "ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC Bioinf.*, vol. 7, p. S7, Mar. 2006.
[44] B. Yang, Y. Chen, and Q. Meng, "Inference of differential equation models by multi expression programming for gene regulatory networks," *Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence* (Lecture Notes in Computer Science), vol. 5755, 2009, pp. 974–983.