Computers in Biology and Medicine 72 (2016) 218–225

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/cbm

TDSDMI: Inference of time-delayed gene regulatory network using S-system model with delayed mutual information

Bin Yang [a,*], Wei Zhang [a], Haifeng Wang [a], Chuandong Song [a], Yuehui Chen [b]

[a] School of Information science and Engineering, Zaozhuang University, Zaozhuang 277160, China
[b] Computational Intelligence Lab, School of Information science and Engineering, University of Jinan, 106 Jiwei Road, 250022 Jinan, China

ARTICLE INFO

Article history:
Received 22 November 2015
Received in revised form
4 March 2016
Accepted 29 March 2016

Keywords:
Time-delayed
Gene expression programming
Gene regulatory network
S-system
Delayed mutual information

ABSTRACT

Regulatory interactions among target genes and regulatory factors occur instantaneously or with time-delay. In this paper, we propose a novel approach namely TDSDMI based on time-delayed S-system model (TDSS) model and delayed mutual information (DMI) to infer time-delay gene regulatory network (TDGRN). Firstly DMI is proposed to delete redundant regulator factors for each target gene. Secondly restricted gene expression programming (RGEP) is proposed as a new representation of the TDSS model to identify instantaneous and time-delayed interactions. To verify the effectiveness of the proposed method, TDSDMI is applied to both simulated and real biological datasets. Experimental results reveal that TDSDMI performs better than the recent reconstruction methods.

1. Introduction

One of the major challenges of system biology is to develop an efficient and accurate framework that could describe the dynamical behaviors of gene regulatory network during recent years [1,2]. Technologies like microarray and high-throughput sequencing could give a large number of gene expression data, which promotes inference of regulatory interactions among genes [3,4].

Gene regulatory network (GRN) is one of the most important biological networks, which has some characteristics, such as strong coupling, random, time-delayed, and strongly nonlinear. In GRN, mRNA of transcription factor must first be translated into protein, and protein regulates the expression of the downstream region of target gene. The process from mRNA to regulation of transcription factor needs a time lag. The actual regulation of transcription factors from the transcription factor to mRNA requires a time delay. Almost all genetic interactions are delayed [5,6]. So time-delayed factor is one of the most important characteristics of gene regulatory network. Many methods have been proposed to identify time-delayed gene regulatory network (TDGRN). Huang et al. proposed GeneReg based on time delay linear regression to construct TDGRN [7]. Li et al. developed a new method that combined relative change ratio and time-delayed dynamic Bayesian network (TDBN) to infer networks [8]. Chueh developed a new approach to reconstruct time delay Boolean networks as a tool for exploring biological pathways [9]. Zoppoli proposed TimeDelay-ARACNE to detect time-delayed dependencies between the expression profiles by assuming as underlying probabilistic model a stationary Markov Random Field [10]. Some preprocess methods were also proposed to determine the optimal time delay in advance. For example, Mundra et al. proposed that an unbiased cross-correlation was used to determine the probability of time delay [11]. ElBakry et al. proposed that pairwise correlations between each pair of genes, for various time delays, were computed and the time delay, at which the correlation was maximum, was the estimated delay between the genes of the pair [12]. Through analysis of these methods, the following problems are not resolved. (1) Due to experimental cost and time, the gene express data has the "large p small n" problem (p is the number of genes and n is the number of experimental sample points). This problem could hinder identification work. (2) The time delay is fixed before the modeling model is constructed.

To achieve a deep understanding of real-world problems in medicine, health care, technology and life sciences, some researchers have proposed reasonable mathematical methods [14]. Mathematical modeling is the art of translating problems from an application area into tractable mathematical formulations, whose theoretical and numerical analysis provides insight, answers, and guidance, useful to understand the original application [13]. One of the most successful mathematical methods is differential equation. The system of differential equations can describe the dynamic properties of a system, which changes with time quite well and

* Corresponding author.
E-mail addresses: batsi@126.com (B. Yang), yhchen@ujn.edu.cn (Y. Chen).

http://dx.doi.org/10.1016/j.compbiomed.2016.03.024
0010-4825/

predict the future states of the system very conveniently. Chen et al. proposed a system of ordinary differential equations (ODEs) to predict the small-time scale traffic measurements data [15]. Wu et al. proposed a sparse additive ODE model, coupled with ODE estimation methods and adaptive group least absolute shrinkage and selection operator techniques, to model dynamic GRNs that could flexibly deal with nonlinear regulation effects [16]. The data from real-word system inevitably contains noise. So instead of the conventional mathematical modeling approaches which are deterministic, stochastic techniques are becoming increasingly necessary. Chowdhury et al. developed a stochastic nonlinear differential equation (S-system) modeling approach to cope with the inherent noise present in the microarray data [17].

In this paper, we propose a new framework based on time delayed S-system (TDSS) model and delayed mutual information (DMI) called TDSDMI to identify TDGRN. TDSDMI contains two steps. One step is that DMI could be used to delete redundant regulator factors and select the proper regulatory factor set for each target gene. This step could decrease the number of candidate regulation genes and revolve the data dimension problem. The second step is that time-delayed S-system (TDSS) model is proposed to infer TDGRN [18]. An improved gene expression programming (GEP), named restricted GEP (RGEP), is proposed as a new representation of the TDSS model. A hybrid evolutionary method based on a structure-based evolutionary algorithm and hybrid particle swarm optimization is used to optimize the architecture and parameters of TDSS model. In the process of optimization of TDSS model, the time delay could be learned from the gene expression data automatically.

One synthetic data and real biological data are used to test the validity of our proposed model TDSDMI. Experimental results demonstrate that our model could infer gene regulatory network accurately.

2. Materials and methods

2.1. Time-delayed S-system (TDSS)

The traditional S-system model possesses a rich structure which can capture various dynamics and has a good compromise between accuracy and mathematical flexibility [19,20]. In order to model the delay characteristic of data, time-delayed S-system (TDSS) model was proposed. The form of each time-delayed differential equation i is given as follows [18]:

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{N} X_{j,t-\tau_{g_{ij}}}^{g_{ij}} - \beta_i \prod_{j=1}^{N} X_{j-\tau_{h_{ij}}}^{h_{ij}}, \quad i = 1 \ldots N. \tag{1}$$

Where $x_{j,t-\tau_{ij}}$ is a vector element of dependent variables, which denotes the expression level of gene X_j at time $t-\tau_{ij}$ ($\tau_{ij} \in [0, \tau_{max}]$). τ_{max} is the predefined maximum delayed length. N is the number of genes, α_i and β_i are vector elements of non-negative rate constants, and g_{ij} and h_{ij} are matrix elements of kinetic orders. When $g_{ij} > 0$, x_j activates the expression of x_i. When $g_{ij} < 0$, x_j suppresses the expression of x_i. h_{ij} is used for degradation.

2.2. Restricted gene expression programming (RGEP)

Ferreira first proposed gene expression programming (GEP) in year 2001 [21]. Due to its linear strings of fixed length like genetic algorithm (GA) and expression as nonlinear entities of different sizes and shapes like genetic programming (GP), GEP and modified GEP have been widely applied in many fields [22–24]. In this paper, according to special form of TDSS model, an improved GEP, named restricted GEP (RGEP), is first proposed.

2.2.1. Chromosome encoding

An example of TDSS represented by RGEP is illustrated in Fig. 1. TDSS contains two terms, so the chromosome in RGEP is composed of two genes. Each gene represents one term. A RGEP gene is a string of function and terminal symbols, which is composed of a head and a tail. The head part contains both function and terminal symbols, whereas the tail part contains terminal symbols only. The function (F) and terminal (T) sets are described as follows:

$$I_1 = F \bigcup T = \{*_1, *_2, *_3, \ldots, *_n\} \bigcup \{x, R\}. \tag{2}$$

Where $*_n$ represents that n variables are multiplied, taking n arguments. x is variable and R is constant.

The head could be created through selecting symbols randomly from the function set F and terminal set T. The symbols of tail are selected from terminal set T only. For each problem, user must determine the head length (h) in advanced. The tail length (t) is computed as:

$$t = (n-1) \times h + 1. \tag{3}$$

Where n is the maximum number of arguments of functions. Suppose that the function set is $\{*_1, *_2, *_3\}$ and the terminal set is $\{x_1, x_2, \ldots, x_5\}$. The length of head is 3, and the length of tail is 7 (the maximum number of arguments of functions n is 3). Fig. 1(a) gives an example of chromosome encoding of RGEP.

The number of genes in RGEP is set as 2, and the link function between two genes is represented by subtraction ($-$). Fig. 1 (b) describes their arithmetic expression tree (ET). The linking function ($-$) connects the expression trees of gene1 and gene2 together to make up the expression tree of one chromosome.

In the process of generating the initial population, three kinds of necessary parameters need be created randomly. The first ones are coefficients α_i and β_i corresponding to gene1 and gene2, respectively. In each gene, kinetic orders (g_{ij} or h_{ij}) are created randomly for every terminal node. The last parameters are time-delayed values (τ_{ij}). Because each sample corresponds to one time stamp in the expression data ($t_1, t_2, \ldots, t_m$ corresponds to $1, 2, \ldots, m$), the time delay parameters are restricted to take only integer values. Each terminal node need to be given two kinds of parameters: kinetic orders and time-delayed values (Fig. 1(a)). The coefficients α_i and β_i, exponents g_{ij} and h_{ij}, and time-delayed values τ_{ij} are optimized by hybrid particle swarm optimization described in Section 2.2.3.

2.2.2. Reproduction

Three genetic operators are used for reproduction of chromosome of RGEP, namely mutation, recombination and selection. Mutation and recombination could generate new offsprings by changing parent population. Selection could select offsprings from parent population according to the fitness. Detailed operating methods are described in [25].

2.2.3. Parameters optimization

To find the optimal coefficients and exponents of RGEP, a hybrid evolutionary algorithm based on particle swarm optimization (PSO) and binary particle swarm optimization (BPSO) is proposed. According to Fig. 1, we check all the parameters ($\alpha_i, \beta_i, g_{i1}, g_{i2}, \ldots, g_{in}, h_{i1}, h_{i2}, \ldots, h_{in}, \tau_{g_{i1}}, \tau_{g_{i2}}, \ldots, \tau_{g_{in}}, \tau_{h_{i1}}, \tau_{h_{i2}}, \ldots, \tau_{h_{in}}$) contained in each model, and count their number N_i ($i = 1, 2, \ldots, M$, M is the population size of TDSS model).

The flowchart is illustrated in Fig. 2. All parameters are encoded into one chromosome. The coefficients (α_i and β_i) and kinetic orders (g_{ij} and h_{ij}) are real numbers, so these two kinds of parameters are optimized using PSO. However the time-delayed values (τ) are integer, so they are optimized by BPSO.

- *Particle swarm optimization*: Each particle x_i represents a