

Applying Two-level Simulated Annealing on Bayesian Structure Learning to Infer Genetic Networks

Tie Wang
Computer Science &
Engineering Department,
Arizona State University,
Tempe, AZ, 85287
tie.wang@asu.edu

Jeffrey W. Touchman
School of Life Science,
Arizona State University,
Tempe, AZ, 85287
j.touchman@asu.edu

Guoliang Xue
Computer Science &
Engineering Department,
Arizona State University,
Tempe, AZ, 85287
xue@asu.edu

Abstract

Bayesian network is a common approach to study gene regulatory networks. Here, we explore the problem of inferring Bayesian structure from data that can be viewed as a search problem. The goal is to find a global optimized probability network model given the data. In this work, we propose a new search algorithm: Two-level Simulated Annealing (TLSA). TLSA performs simulated annealing in two levels with strengthened local optimizer, and is less likely to get tracked at local optimizer. To illustrate the value of TLSA in Bayesian structure learning, the algorithm is applied on simulated datasets generated using the Monte Carlo method. The experimental results are compared with other learning algorithm such as K2.

1. Introduction

Recent high-throughput molecular biology has motivated the development of algorithms and tools for analyzing gene expression data. The central goal is to understand the regulatory mechanism between gene-gene, protein – gene, and protein – protein. Inference of gene networks has been primary focus of research.

There are several gene expression data analysis tools used to reveal the regulatory relations among genes, such as clustering and differential equations. Although clustering algorithms can successfully reveal co-regulated genes, they can not find regulatory relationships [2]. At the same time, the detail requirement of relation and parameters of biochemical reactions restrict differential equations to very small systems. Bayesian networks are widely used to infer gene regulatory network from expression data. Bayesian network is a graphical representation of probabilistic relationships between multiple variables. Compared with other methods, Bayesian network is

resistant to noise in data, making it more robust for inferring structure. In this paper, we propose an optimization based algorithm to infer the Bayesian network structure. We compare the experimental results with the K2 algorithm based on a Bayesian score value. The results and analysis are provided in section 3.

2. Method

Inferring Bayesian structure from expression data can be viewed as a search problem in the network space. The goal is to find an optimized network model for the data with maximized/minimized score. Heuristic search algorithms with/without ordering are applied to the structure learning problem because given prior knowledge and the data, the search problem is known to be NP-hard. Simulation annealing is a combinatorial optimization algorithm which is an extension of a Monte Carlo method to determine the equilibrium states of a collection of atoms at any given temperature T [1]. It has been proved very successful on solving combinatorial optimization problems including Bayesian structure inferring. The drawback of the algorithm is that after a large number of iterations, the temperature drops to a low degree and the local optimizer reaches its stable state. That means even after applying a perturbation on the optimizer, the new search solution is very likely to fall into the same basin. In such a case, the search will stop at a local optimized solution. In our two-level simulated annealing (TLSA) [1], we can solve this optimization problem. The algorithm is described as follows:

Algorithm: two-Level Simulated Annealing

Set T to its initial value. Set x_{old} to an initial feasible solution. Compute x_{old}' . Set $f_{old} = f(x_{old}')$; $x_{best} = x_{old}$ and $f_{best} = f_{old}$.

Repeat

For $i=1$ to m (m is the number of iterations)
 $x_{new} = \text{perturbation}(x_{old}); f_{new} = f(x_{new}')$;
generate a random number r (0,1);
if($(f_{new} < f_{old})$ or ($r \leq \exp(f(x_{old}) - f(x_{new}))/T$)) then
 $x_{old} = x_{new}; f_{old} = f_{new}$;
if($f_{old} \leq f_{best}$) then
 $x_{best} = x_{new}; f_{best} = f_{new}$;

$T = p * T$;

until (stopping criteria is met)

x_{best}' is the optimized solution and f_{best} is the best objective function value. [1]

In the TLSA, we set up two levels for each candidate x and named them upper level and lower level. The objective function value of lower level for x is the local optimizer value. The perturbation is made on the upper level while the decision on accepting or rejecting the move is depend on the comparison result of lower level objective function values of current and new points. The TLSA could look ahead for the objective function info on local optimizer before making any decision. In the case of low temperature, the TLSA still could work very effectively by accepting better local optimizer without being trapped into certain local optimizer basins.

We can apply TLSA in Bayesian structure learning problem. The x in TLSA algorithm represents a DAG (Directed Acyclic Graph) pattern of Bayesian network. From the decomposition rule in Bayesian network, we can apply Bayesian score equation as the objective function for each variable (gene).

$$\begin{aligned} S(G:D) &= \log P(G | D) \\ &= \log P(D | G) + \log P(G) + C \\ S_{BDe} &= \sum_i \text{Score}_{BDe}(X_i, \text{Pa}(X_i) : D), \end{aligned}$$

where G is the DAG, D is the complete data and X_i is the variable(gene) in the network, $\text{Pa}(X_i)$ is the parent set of variable X_i .

The local neighborhood in TLSA could be defined by the operations on the edge between any two nodes in G . The operations between any two nodes are: adding edge, deleting edge and reversing edge. Given certain x which is a network structure, we can apply $O(n^2)$ single-step operations in each that will generate x_{new} in the neighborhood of x . By applying m -step ($m > 1$) operations simultaneously, we could move to another disjoint neighborhood for further search.

3. Results and Conclusion

We applied TLSA algorithm to simulated data set. We build up “Golden Networks” (GNs) with 10 nodes, 20 nodes, 30 nodes and 40 nodes respectively. Simulated data sets are generated from GNs by applying the Monte Carlo Method. We use the resultant sampled data to test Bayesian scores that infer the strength of learning network structures in the design of the TLSA algorithm. The data in each data set has value 0 or 1 which represents down-regulated and up-regulated respectively. In our experiment, the sample size is also fixed. Simulation is run on 10 networks for each kind of GN, the minimized scores are compared with the score we obtained using K2 algorithm. The results are listed in the following graphics:

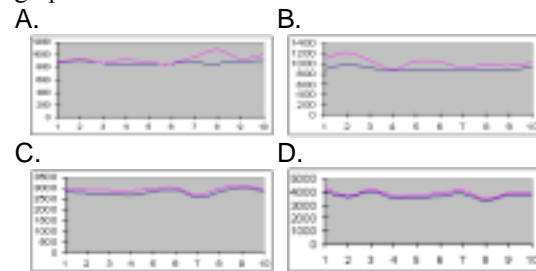


Fig 3.1 Score graphs between TLSA and K2: A-D represent Bayesian scores on networks with 10,20,30,40 nodes respectively. Top line represents TLSA score and bottom line represents K2 score.

The simulation results show that TLSA can reach better structure with lower score compared with K2 although no ordering information is available in advance. Therefore, TLSA is more likely to find equivalent pattern of the optimized structure from data. Analysis of regulatory networks using other data as well as gene expression data is currently underway.

Acknowledgement:

This research was supported in part by ARO grant DAAD19-00-1-0377 (to Xue).

Reference

- [1] Guoliang Xue, “Parallel Two-Level Simulated Annealing”, ICS’93, ACM, Tokyo, Japan, 1993.
- [2] Nir Friedman Michal Linial, Iftach Nachman, Dana Pe’er, “Using Bayesian Networks to Analyze Expression Data”, J. Comput Bio, vol7, 601-620, 2000.
- [3] Smith VA, Jarvis ED, Hartemink AJ, “Evaluating functional network inference using simulations of complex biological systems” *Bioinformatics*, 18, S216–S224, 2002.