A Parallel Algorithm for Estimating Genome-Wide Gene Networks using Nonparametric Bayesian Networks

Yoshinori Tamada\(^1\) Seiya Imoto\(^1\)
tamada@ims.u-tokyo.ac.jp imoto@ims.u-tokyo.ac.jp

Hiromitsu Araki\(^2\) Masao Nagasaki\(^1\) Satoru Miyano\(^1\)
hiromitsu_araki@cell-innovator.com masao@ims.u-tokyo.ac.jp miyano@ims.u-tokyo.ac.jp

\(^1\) Human Genome Center, Institute of Medical Science, University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, Tokyo 108-8639, Japan
\(^2\) Cell Innovator Inc, Fukuoka, Japan 812-8581

Keywords: genome-wide gene network estimation, nonparametric Bayesian networks

1 Introduction

We present a novel algorithm for estimating genome-wide gene networks using nonparametric Bayesian network models [3]. The algorithm, which is called the Neighbor Node Sampling & Repeat (NNSR) algorithm, is capable of searching a Bayesian network structure consisting of more than 20,000 nodes, which is fitted to given gene expression data. To realize the large scale Bayesian network structure search, the algorithm is designed to run on massively parallel computers where a massive amount of independent computation nodes are linked by fast connection. Such a computer system is also known as a distributed-memory architecture supercomputer. A Bayesian network is widely used as a gene network model [2]. Learning of a Bayesian network structure from gene expression data, however, is known as an NP-hard problem, and therefore the optimal network structure can be estimated only up to less than 30 genes or fewer. Thus, for the larger network, a heuristics algorithm such as the greedy hill-climbing (HC) algorithm is often used. The HC algorithm is applicable to networks with up to 1000 genes. A gene network with 1000 genes, however, involves fewer than 5% of all human genes. Therefore, the current Bayesian network estimation technology is far from a genome-wide scale analysis for most species.

2 Algorithm

A naïve way to deal with such a large number of genes during the network estimation is to repeatedly estimating smaller subnetworks, where each subnetwork incorporates randomly sampled subsets of genes. This enables us to perform each subnetwork estimation completely independently on each computational node. However, simple random sampling of genes requires a very large number of iterations to cover all possible pairs of nodes. To overcome this problem, our NNSR algorithm performs neighbor node sampling that consists of a random walk on a network with reliable edges estimated during the algorithm execution, and the calculation of 1-to-1 edge scores derived from a local network score, in order to help select which genes should be included together for the single subnetwork estimation. This significantly improves the accuracy of the network estimation and requires fewer iterations. The final network structure is determined by calculating the frequencies of edges estimated during the repeated subnetwork estimation. Although this does not generate a DAG structure, it is enough for biological analysis in terms of the gene network analysis. The exact DAG structure can also be obtained by taking edges according to the edge frequencies with keeping the structure acyclic.
3 Simulation Results

We evaluated the performance of the algorithm using simulated expression data. At first, we tested the speedup efficiency of the algorithm. Fig. 1 (left) shows the result of the speedup efficiency test. We measured the running times using 8, 16, 32, 64, 128, 256, and 512 processes using 1, 2, 4, 8, 16, 32, and 64 computation nodes where each has two quad core Intel 3.0 GHz Xeon processors. The speedup efficiency with 512 processes against 8 processes was 0.89, which shows that our algorithm has high scalability to the number of processes (136.29 sec with 8 processes → 18.88 sec with 512 processes).

Next we compared the estimated networks with several parameter settings and the results of the HC algorithm. Fig. 1 (right) represents the ROC curves for the specificity and the sensitivity to the true structure with various final cutoff values for the edge frequencies. Note that the results of the proposed algorithm are not DAGs except for (k), (l) and (m). The long HC execution over 40 hours (n) recorded the best result. However, short time execution of our NNSR algorithm with 8 minutes post processing by HC outperformed any other results. This represents the high applicability of our algorithm to large scale gene network estimation.

References

