Inferring *E. coli* SOS Response Pathway from Gene Expression Data Using IST-DBN with Time Lag Estimation

Lian En Chai, Mohd Saberi Mohamad*, Safaai Deris,
Chuii Khim Chong, and Yee Wen Choon

Artificial Intelligence and Bioinformatics Research Group, Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia
{lechai2,ckchong2,ywchoon2}@live.utm.my, {saberi,safaai}@utm.my

Abstract. Driven to discover the vast information and comprehend the fundamental mechanism of gene regulations, gene regulatory networks (GRNs) inference from gene expression data has gathered the interests of many researchers which is otherwise unfeasible in the past due to technology constraint. The dynamic Bayesian network (DBN) has been widely used to infer GRNs as it is capable of handling time-series gene expression data and feedback loops. However, the frequently occurred missing values in gene expression data, the incapability to deal with transcriptional time lag, and the excessive computation time triggered by the large search space, are attributed to restraint the effectiveness of DBN in inferring GRNs from gene expression data. This paper proposes a DBN-based model (IST-DBN) with missing values imputation, potential regulators selection, and time lag estimation to address these problems. To assess the performance of IST-DBN, we applied the model on the *E. coli* SOS response pathway time-series expression data. The experimental results showed IST-DBN has higher accuracy and faster computation time in recognising gene-gene relationships when compared with existing DBN-based model and conventional DBN. We also believe that the ensuing networks from IST-DBN are applicable as a common framework for prospective gene intervention study.

Keywords: Dynamic Bayesian network, missing values imputation, time-series gene expression data, gene regulatory networks, network inference.

1 Introduction

In the post-genomic era, aided by the breakthroughs in technology, researchers have begun to shift the research paradigm from the classical reductionism to the modern holism, wherein biological systems and experimental design are viewed as a whole instead as collections of parts [1]. One of the innovations conceived in such era, the

[*] Corresponding author.

A.S. Sidhu & S.K. Dhillon (Eds.): *Adv. in Biomedical Infrastructure 2013*, SCI 477, pp. 5–14.
DOI: 10.1007/978-3-642-37137-0_3

DNA microarray technology, which is capable of representing the expression of thousands of genes under various circumstances (otherwise known as gene expression profiling), has allowed the development of numerous new experiments for exploring into the complex system of gene expression and regulation [2]. Since its conception, various organisms and mammalian cells have been profiled, such as *S. cerevisiae* [3], human cancerous tissue [4], and *E. coli* [5]. The consequent output, commonly known as gene expression data, comprises immense information such as the robustness and behaviours denoted by the cellular system under diverse situations [6], assists us in understanding the underlying mechanism of gene expression and regulation.

From a computational perspective, a GRN can be represented as a directed graph containing nodes (genes) and edges (interaction/relationship). In recent years, various computational methods have been developed to infer GRNs from gene expression data. Among them, Bayesian network (BN) [7], which uses probabilistic correlation to distinguish relationships between a set of variables, was popular in GRNs inference. This is mainly due to several factors: BN is capable of working on local elements, assimilating other mathematical models to avert data overfitting, and merging prior knowledge to fortify the causal relationships. Nonetheless, BN also has two disadvantages: it is unable to deal with time-series gene expression data and construct feedback loops.

From a biological perception, feedback loops actually embody the homeostasis procedure in living organisms. Hence, to take account of the feedback loops, researchers have developed the dynamic Bayesian network (DBN) [8] as a replacement to tackle BN's weaknesses. However, the scattering missing values commonly found in gene expression data could affect more than 90% of the genes and subsequently negatively influencing downstream analysis and inferring approaches [9]. Furthermore, in identifying gene-gene relationships, conventional DBN generally comprises all genes into the subsets of potential regulators for each target gene, and thus instigated the large search space and the excessive computational time [10]. To address the two problems, Chai *et al.* [11] suggested a three-step DBN-based model (ISDBN) with missing values imputation and potential regulators selection, and the proposed model showed better performance than conventional DBN in GRNs inference.

Yet, ISDBN and conventional DBN is still not adept enough to effectively take account of the transcriptional time lag, in which a time delay exists before the target genes are being expressed into the system. This shortcoming hampers the accuracy of DBN-based approaches in GRNs inference. To solve this problem, we proposed to further improve the aforesaid DBN-based model with time lag estimation (IST-DBN) which would take account of the transcriptional time lag based on the time difference between the initial changes of expression level of potential regulators and their target genes.

2 Methods

Essentially, IST-DBN involves four main steps: missing values imputation, potential regulators selection, time lag estimation and DBN inference. Fig. 1 illustrates the schematic overview of IST-DBN.

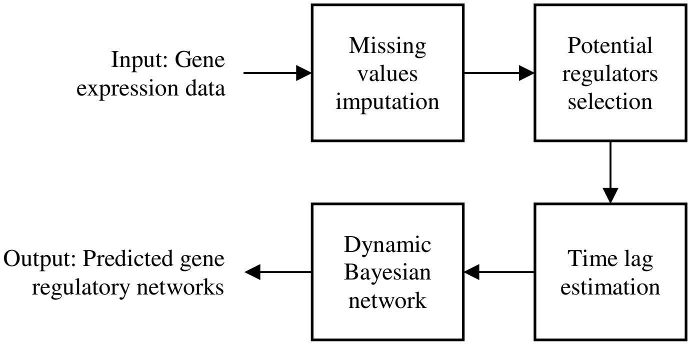
Input: Gene expression data
Missing values imputation
Potential regulators selection
Time lag estimation
Dynamic Bayesian network
Output: Predicted gene regulatory networks

Fig. 1. Schematic overview of IST-DBN

2.1 Missing Values Imputation

Missing values in gene expression data can occur for numerous reasons. For example, small contaminations would corrupt the microarray slides at multiple spots as they are very tiny and packed together. These questionable spots are then labelled as missing after scanning and digitalising the microarray slides. Many imputation methods have been established to impute missing values by exploring and utilising the underlying expression data structure and pattern. In particular, based on the local similarity structure, LLSimpute imputes missing values by constructing a linear combination of similar genes and target genes with missing values through a similarity measure [12]. This method entails two steps. Firstly, k genes are selected by the L_2-norm, where k is a positive integer that expresses the number of coherent genes to the target gene. As an example, to impute a missing value g found at $\boldsymbol{x}_{11}$ in a $m\times n$ matrix $\boldsymbol{X}$, the k-nearest neighbour gene vectors for $\boldsymbol{x}_1$,

$$\boldsymbol{v}_{s_i}^{\mathrm{T}} \in \boldsymbol{X}^{1\times n} \quad 1 \le i \le k \tag{1}$$

are computed, whereby the gene expression data is defined as a $m\times n$ matrix $\boldsymbol{X}$ (m is the number of genes, n is the number of observations), and $\boldsymbol{x}_1$ signifies the row of the first gene with n observations. s_i is a list of k-nearest neighbour genes vectors, which actually corresponds to the i-th row of the transpose vector $\boldsymbol{v}^{\mathrm{T}}$. The following step implicates regression and estimation of the missing values. A matrix, $\boldsymbol{A} \in \boldsymbol{X}^{k\times(n-1)}$ wherein the k rows of the matrix contains vector $\boldsymbol{v}$, and two vectors, $\boldsymbol{b} \in \boldsymbol{X}^{k\times 1}$ and $\boldsymbol{w} \in \boldsymbol{X}^{(n-1)\times 1}$, are then formed. The vector $\boldsymbol{b}$ encloses the first element of k vectors $\boldsymbol{v}^{\mathrm{T}}$, whereas vector $\boldsymbol{w}$ comprises $n - 1$ elements of vector $\boldsymbol{x}_1$. A k-dimensional coefficient vector $\boldsymbol{y}$ is subsequently computed such that the least square problem is minimised as

$$\min_{\boldsymbol{y}} |\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{w}|^2 \tag{2}$$

Let $\boldsymbol{y}^*$ to denote the vector wherein the square is minimised such that

$$\boldsymbol{w} \simeq \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}^* = \boldsymbol{y}_1^*\boldsymbol{a}_1 + \boldsymbol{y}_2^*\boldsymbol{a}_2 + \cdots + \boldsymbol{y}_k^*\boldsymbol{a}_k \tag{3}$$

where $\boldsymbol{a}_i \in \boldsymbol{A}^{k\times 1}$, and thus, the missing value g could be imputed as a linear combination of coherent genes such that

$$g = \boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{b}^{\mathrm{T}}(\boldsymbol{A}^{\mathrm{T}})'\boldsymbol{w} \tag{4}$$

where $(\boldsymbol{A}^{\mathrm{T}})'$ exists as the pseudoinverse of $\boldsymbol{A}^{\mathrm{T}}$ [12].

2.2 Potential Regulators Selection

In most occurrences, the expression level of regulators (also known as TFs, transcriptional factors) would vary before or simultaneously with their target genes [13]. By exploiting this information, we formulated an algorithm which would shrink the search space by confining the number of potential regulators for each target gene. Firstly, a threshold for categorising the status of gene expression values (e.g. up- or down-regulation) is determined through either experiments or the average expression level of the genes. In this paper, the threshold for up-regulation and down-regulation are decided based on the baseline cut-off of the gene expression values. As such, for the *E. coli* dataset used in this paper, the threshold is determined as ≥1.4 for up-regulation and ≤0.7 for down-regulation. The gene expression values are successively categorised into one of the three states: up-, down- and normal regulation. The three states specify whether the expression value is greater than, lower than or similar to the threshold. Subsequently, the precise time units of initial up-regulation and down-regulation of each gene are chosen, and genes with preceding fluctuations in expression level are encompassed into the subset of potential regulators against genes with later expression fluctuations. As genes with significantly late expression fluctuations could have involved a large number of potential regulators, the maximum time gap for preceding expression fluctuations is constrained to five time units. This is to avert choosing potential regulators for a target gene from the entire gene expression dataset. To further elucidate this algorithm, let's assume two hypothetical genes: gene *P* and gene *R*. Gene *P* encountered an initial expression change at time T_1 prior to the initial expression change of gene *R* at time T_2, hence gene *P* is included into the subset of potential regulators for gene *R* (Fig. 2). The same procedure applies to other up- or down-regulated genes which satisfy the criteria.

2.3 Time Lag Estimation

Transcriptional time lag is the time interval between the expression of the regulators and the expression of their target genes. Remember the two hypothetical genes, *P* and *R*,

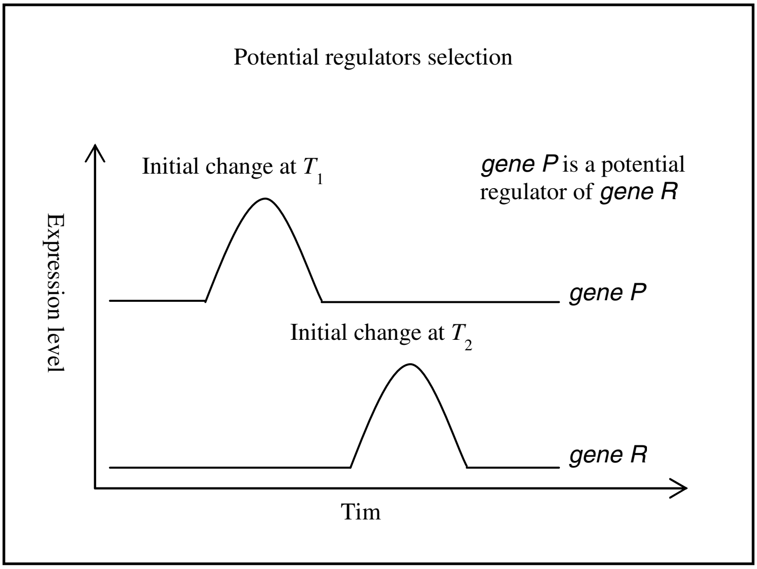

Fig. 2. Schematic overview of potential regulators selection

whereby gene *P* regulates gene *R*. Gene *P* starts expression change at time T_1 and gene *R* has an expression change at T_2. The time difference between T_1 and T_2 is regarded as the transcriptional time lag. In inferring GRNs from gene expression data, conventional DBN aligns regulator-gene pairs based on the statistical analysis of their probabilistic strength between time units. Nonetheless, DBN usually pairs up regulators with their target genes by only one time unit, although the actual transcriptional time lag could have been multiple time units. With such cases, IST-DBN takes consideration of the real transcriptional time lag by coupling up potential regulators and their target genes based on the time difference between their initial expression fluctuations. For a target gene, potential regulators are categorised into different groups based on the time lag (e.g. groups of one, two or three time units), mostly due to the fact that a target gene could have numerous regulators acting upon it in dissimilar time unit.

2.4 Dynamic Bayesian Network

DBN infers time-series gene expression data by observing the values of a set of variables at diverse time units. DBN inference typically involves two steps: parameter learning and structure learning. In parameter learning, the joint probability distribution (JPD) of the variables is calculated based on the Bayes theorem. Let's assume a microarray dataset with m genes and n observations, such that we have a $m \times n$ matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ wherein each row, vector $\boldsymbol{x}_m = (\boldsymbol{x}_{m1}, \ldots, \boldsymbol{x}_{mn})$ embodies a gene expression vector observed at time t. The temporal vectors chain relationship is defined as a *first-order Markov chain* in which only forward edges are permitted. The JPD of the model has the overall form of:

$$P(\boldsymbol{x}_{11}, \dots, \boldsymbol{x}_{mn}) = P(\boldsymbol{x}_1)P(\boldsymbol{x}_2|\boldsymbol{x}_1) \dots P(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}) \quad (5)$$

Based on the earlier threshold, the expression values acquired from preceding steps are discretised into three categories: -1, 0 and 1, which correspond to down-, normal and up-regulation respectively. Each set of potential regulators is subsequently distributed into smaller subsets. For instance, in a set of potential regulators comprising gene X, and gene Y, the subsets would be $\{X\}$, $\{Y\}$ and $\{X, Y\}$. Each of the subset and the target gene are then arranged into a data matrix with their discretised expression values. The conditional probabilities of each subset of potential regulators against their target genes are then calculated. The following step is to look for the optimal network structure through a scoring function based on the Bayesian Dirichlet equivalence (BDe). The final results are then imported into GraphViz (*http://www.graphviz.org*) for network visualisation and analysis.

3 Result and Discussion

3.1 Experimental Data and Setup

The experimental data involved in this paper is the *E. coli* SOS response pathway gene expression data [14]. The *E. coli* SOS response pathway is a DNA restoration system which reacts to damaged DNA by pausing cell cycle and triggering DNA repair [15]. In normal situation, the SOS genes are negatively regulated through the binding of the repressor protein, lexA to the promoter region of these genes. When DNA is damaged, DNA polymerase is blocked and single-stranded DNA (ssDNA) start to accumulate. The sensor of DNA damage, the recA protein, activates by binding to these ssDNA. After being activated, the recA protein initiates the self-cleavage of the lexA repressor. This would cause a drop in lexA level and in turn the SOS genes are de-repressed. This remains until the damage is restored, wherein the level of activated recA falls, lexA amasses and represses the SOS genes again. This dataset comprises 8 genes observed at uniformly spaced 50 time units with 6 minutes apart, and also 11.5% missing values (184 out of 1,600 observations).

The DBN inferring part of IST-DBN is applied under the framework of BNFinder [16], while the missing values imputation, potential regulators selection and the time lag estimation are applied in MATLAB environment. To assess the performance of IST-DBN, the accuracy and computation time of the proposed model is compared against ISDBN and DBN (characterised by BNFinder). The accuracy is evaluated by comparing the results of the three models to the reputable *E. coli* SOS response pathway by Ronen *et al.* [14]. All three models are executed using the same hardware configuration (3.2GHz Intel Core i3 computer with 2GB main memory) to ensure a fair assessment of computation time. Table 1 summarises the results, wherein the first row denotes the network inferred by IST-DBN, the second row denotes the network inferred by ISDBN, and the third row denotes the network inferred by DBN. An edge shows a relationship between the two linked genes. 'Correctly inferred relationships' represents the number of relationships which are found in both inferred and established networks, 'sensitivity' is the rate of correctly inferred relationships, and 'specificity' relates to the rate of correct inference that no relationship exists between two genes.

3.2 Experiment Results

IST-DBN succeeded in identifying all ten relationships (lexA–recA, lexA–polB, lexA–umuD, lexA–uvrY, lexA–uvrA, lexA-uvrD, lexA–ruvA, lexA–lexA, recA–recA, and recA–lexA) (Fig. 3), whereby ISDBN correctly identified nine relationships and DBN only recognised eight relationships. Both IST-DBN and ISDBN outperformed DBN in this category, and this is because the effectiveness of DBN was hampered by numerous missing values in the original gene expression data. Also, through the alignment of regulator-gene pairs based on actual transcription time lag, the causal correlation between pairs with greater transcriptional time lag are strengthened, due to the fact that IST-DBN reported lesser false positives when compared with ISDBN and DBN (3 against 6 and 5). IST-DBN registered 100% sensitivity and 83.33% specificity compared to ISDBN's 90% sensitivity and 66.67% specificity. Conversely, DBN reported 80% sensitivity and 72.22% specificity. The perfect sensitivity of IST-DBN and the relatively significant difference in percentage with the other two models is obviously attributed to the relatively small dataset, but we expect that IST-DBN would still outperform ISDBN and DBN on larger dataset. All three models were capable of identifying at least two self-regulatory loops: recA, which senses DNA damage and subsequently self-activate by binding to ssDNA; and lexA, which undergoes self-cleavage after initiated by the relatively high level of activated recA.

Four probable situations arise when an edge exist between two genes: correct direction and regulation type, correct direction but incorrect regulation type, misdirected but correct regulation type, and misdirected and wrong regulation type. IST-DBN was able to revise an incorrect relationship type in ISDBN. However, IST-DBN also contains an incorrect regulation type while ISDBN showed two wrong regulation type and one misdirected edges, and conventional DBN reported three incorrect regulation type and one misdirected edges. In regard to the computation time, IST-DBN demonstrated a computation time of 7 minutes and 56 seconds while ISDBN showed a computation time of 8 minutes and 43 seconds. On the contrast, DBN recorded 15 minutes and 17 seconds. As the dataset used in this study was relatively small, the computation time for IST-DBN and ISDBN do not differ drastically, although DBN suffers from longer computation time which is caused by a larger search space. We expect that the computation time difference between DBN and the other two models would be much more radical with a larger dataset.

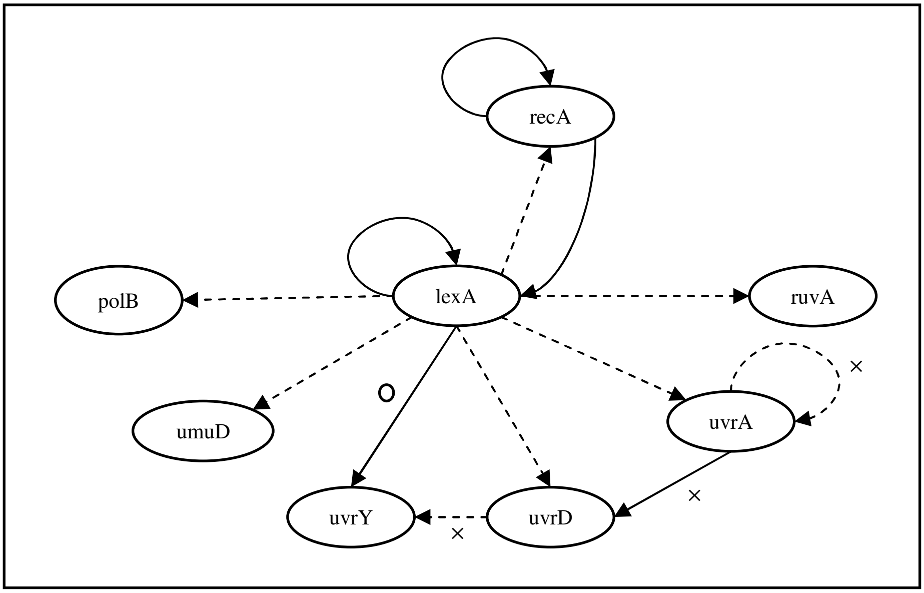

Fig. 3. Inferred *E. coli* SOS response pathway using IST-DBN. Dash edges (---) indicate down-regulations and straight-lined edges (—) indicate up-regulations. A cross denotes an incorrect inference; a circle denotes an incorrect regulation type; an edge without any attachment is a correct inference.

Table 1. The results of experiment study

Model	Correctly predicted relationships	Sensitivity	Specificity	Computation time (HH:MM:SS)
IST-DBN	10	100.00%	83.33%	00:07:56
ISDBN	9	90.00%	66.67%	00:08:43
DBN	8	80.00%	72.22%	00:15:17

4 Conclusion

Traditional DBN has been troubled by three main problems: the missing values commonly found in gene expression data, the comparatively large search space due to encompassing all genes as potential regulators against target genes, and the absence of a method to consider transcriptional time lag. ISDBN was put forth by Chai *et al.* [11] to tackle the first two problems: Missing values are imputed based on linear grouping of analogous genes, and the search space is diminished by restricting to certain potential regulators which fulfill the criteria. Nevertheless, this model is unable to deal with transcription time lag and thus, we proposed an enhanced version of ISDBN with time lag estimation (known as IST-DBN) to solve the third problem. Rather than pairing up with the default one time unit, IST-DBN utilises the actual time difference between expression changes to align regulator-gene pairs. Therefore, IST-DBN is capable of seizing most of the probabilistic connection between genes that possess transcriptional time lag greater than one time unit. Based on the *E. coli* SOS response pathway dataset, IST-DBN presented encouraging results in regards to accuracy and computation time when matched against ISDBN and traditional DBN. We are interested to apply IST-DBN to other datasets, for instance, *S. cerevisiae* or *A. thaliana*, to examine the performance consistency of IST-DBN.

Acknowledgments. We would like to thank the Malaysian Ministry of Science, Technology and Innovation for supporting this research by an e-science research grant (Grant number: 06-01-06-SF1029). This research is also supported by UTM GUP research grant that was sponsored by Universiti Teknologi Malaysia.

References

1. Lee, W.P., Tzou, W.S.: Computational methods for discovering gene networks from expression data. Brief Bioinform. 10(4), 408–423 (2009)
2. Jornsten, R., Wang, H.Y., Welsh, W.J., Ouyang, M.: DNA microarray data imputation and significance analysis of differential expression. Bioinformatics 21(22), 4155–4161 (2005)
3. Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B.: Comprehensive identification of cell cycle regulated genes of the yeast *Saccharomyces* cerevisiae by microarray hybridization. Mol. Biol. Cell 9, 3273–3297 (1998)
4. Muro, S., Takemasa, I., Oba, S., Matoba, R., Ueno, N., Maruyama, C., Yamashita, R., Sekimoto, M., Yamamoto, H., Nakamori, S., Monden, M., Ishii, S., Kato, K.: Identification of expressed genes linked to malignancy of human colorectal carcinoma by parametric clustering of quantitative expression data. Genome Biol. (4), 21 (2003)
5. Salgado, H., Gama-Castro, S., Peralta-Gil, M., Diaz-Peredo, E., Sanchez-Solano, F., Santos-Zavaleta, A., Martinez-Flores, I., Jimenez-Jacinto, V., Bonavides-Martinez, C., Segura-Salazar, J., Martinez-Antonio, A., Collado-Vides, J.: RegulonDB (version 5.0): *Escherichia coli* K-12 transcriptional regulatory network, operon organization, and growth conditions. Nucleic Acids Res. 34, 394–397 (2005)
6. Karlebach, G., Shamir, R.: Modelling and analysis of gene regulatory networks. Nat. Rev. Mol. Cell Bio. 9(10), 770–780 (2008)
7. Friedman, N., Linial, M., Nachman, I., Pe'er, D.: Using Bayesian networks to analyse expression data. J. Comp. Biol. 7, 601–620 (2000)
8. Murphy, K., Mian, S.: Modelling gene expression data using dynamic Bayesian networks. Technical report, Computer Science Division, University of California, Berkeley (1999)
9. Ouyang, M., Welsh, W.J., Geogopoulos, P.: Gaussian mixture clustering and imputation of microarray data. Bioinformatics 20(6), 917–923 (2004)
10. Jia, Y., Huan, J.: Constructing non-stationary dynamic Bayesian networks with a flexible lag choosing mechanism. BMC Bioinformatics (11), 27 (2010)
11. Chai, L.E., Mohamad, M.S., Deris, S., Chong, C.K., Choon, Y.W., Ibrahim, Z., Omatu, S.: Inferring gene regulatory networks from gene expression data by a dynamic bayesian network-based model. In: Omatu, S., De Paz Santana, J.F., González, S.R., Molina, J.M., Bernardos, A.M., Rodríguez, J.M.C. (eds.) Distributed Computing and Artificial Intelligence. AISC, vol. 151, pp. 379–386. Springer, Heidelberg (2012)

12. Kim, H., Golub, G., Park, H.: Missing value estimation for DNA microarray gene expression data: local least squares imputation. Bioinformatics 21(2), 187–198 (2005)
13. Yu, H., Luscombe, N.M., Qian, J., Gerstein, M.: Genomic analysis of gene expression relationships in transcriptional regulatory networks. Trends Genet. 19, 422–427 (2003)
14. Ronen, M., Rosenberg, R., Shraiman, B.I., Alon, U.: Assigning numbers to the arrows: parameterizing a gene regulation network by using accurate expression kinetics. Proc. Natl. Acad. Sci. 99, 10555–10560 (2002)
15. Radman, M.: Phenomenology of an inducible mutagenic DNA repair pathway in *Escherichia coli*. Basic Life Sci. 5A, 255–367 (1975)
16. Wilczynski, B., Dojer, N.: BNFinder: exact and efficient method for learning Bayesian networks. Bioinformatics 25(2), 286–287 (2009)