Inferring Regulatory Pathways in E. Coli using Dynamic Bayesian Networks

Irene M. Ong
Dept. of Computer Sciences
University of Wisconsin
Madison, WI 53703

David Page
Dept. of Biostatistics & Medical Informatics
Dept. of Computer Sciences
University of Wisconsin
Madison, WI 53703

Abstract

This paper applies Dynamic Bayes Nets to the task of predicting gene expression in *E. coli*. Specifically, the evidence variables in our DBN are discretized gene expression levels, and the hidden state variables are "operon" transcription levels. An operon is a sequence of genes that are transcribed together; our operons include both known operons from [Salgado, Moreno-Hagelsieb, Smith and Collado-Vides 2000] and predicted operons from [Craven, Page, Shavlik, Bockhorst and Glasner 2000]. The arcs from state to evidence variables are known, but the arcs from state variables of one time step to state variables of the next time step are unknown. These arcs, as well as all CPT probabilities, are inferred from time-series microarray data for *E. coli*.

1 Introduction

Over the past decade, scientists have been rushing to produce complete genome sequences for various organisms ranging from microbes to humans. These completed genomes are springboards to the ultimate goal of understanding the workings of complex living systems. Each genome contains thousands of genes, each of which codes for one or more proteins; these proteins may in turn regulate other genes through complex regulatory pathways to accomodate changes in their environment or carry out the programs of growth and development of the organism. The key to understanding living processes is uncovering this genome-wide circuitry that underlie the regulation of cells.

The laborious task of identifying metabolic pathways through many experiments over the last century has been fundamental for drug development. Discovering transcriptional regulatory pathways will have an even greater impact on medicine because regulatory pathways that fail to function correctly are the major cause of ailments such as cancer (due to uncontrolled cell growth).

To uncover genome-wide regulatory pathways, we need a way to measure the expression levels of all the genes within the organism. DNA microarrays enable the simultaneous measurement of most if not all identified genes in a genome. However, the data from microarrays are inherently noisy; our approach to analysis uses both discretization (a common technique) and background knowledge to partially offset the noise.

We introduce an approach to determining transcriptional regulatory pathways by applying Dynamic Bayes Nets to time-series gene expression data from DNA microarray hybridization experiments. Our approach involves building an initial DBN structure that exploits background knowledge of operons and their associated genes. We use the operon map from [Craven, Page, Shavlik, Bockhorst and Glasner 2000] that maps every known and putative gene in the *E. coli* genome into its most probable operon. This map makes the simplifying assumption (rarely but occasionally violated) that every gene appears in exactly one operon, and its accuracy is estimated at about 95%. We also use our best guess of priors for setting initial conditional probability tables (CPTs) in the DBN. The structural EM algorithm [Friedman 1998] is used to infer the remaining structure of the DBN from *E. coli* gene expression data, with the EM algorithm revising the CPTs. The use of prior knowledge—operons and prior CPTs—is especially important because of the noise and small sample size of the data.

[Friedman, Linial, Nachman and Pe'er 2000] were the first to address the task of determining properties of the transcriptional program of an organism (Baker's yeast) by using Bayesian Networks to analyze gene expression data. Their method can represent the dependence between interacting genes, but it does not show how genes regulate each other over time in the complex workings of regulatory pathways. Analysis of time-series data potentially allows us to determine regulatory pathways across time rather than just associating genes that are regulated together.

To our knowledge [Friedman, Linial, Nachman and Pe'er 2000] and [Murphy and Mian 1999] are to be credited with first proposing the suitability of DBNs for modeling time-series gene expression microarray data. [Murphy and Mian 1999] cited the advantages of DBNs as being stochasticity and the abilities to incorporate prior knowledge and hidden variables. The primary contribution of our paper is to test this DBN approach on real time-series microarray data. A secondary contribution is the incorporation of the results of a previous application of Bayesian inference (naive Bayes) as background knowledge for this new application. The previous application used a variety of evidence sources, including earlier microarray data from the Blattner Laboratory at the University of Wisconsin, to predict the operons in *E. coli*. The goal of that work was to produce an accurate operon map that could later be used in the prediction of regulatory pathways in *E. coli*. The present paper describes a next step in this direction.

The experiments described in the following section are designed to test two hypotheses. The first is that DBN structure learning will induce arcs between operons that are involved in the same regulatory pathway. If this hypothesis proves to be true, then DBN structure learning can propose links that investigators can further pursue by experimentation. The second, stronger hypothesis is that DBN structure learning will induce arcs that *actually appear* in the regulatory pathway. This second hypothesis is much stronger since the actual arcs might not be induced for a variety of reasons including: the time steps in the data set are too large, the data are too noisy, or the learning algorithm recognizes the dependence but not the "causality" (the arc is in the wrong direction). The next section provides evidence that supports the first hypothesis but not the second. Section 3 discusses further work that might be done to improve the DBN learning approach and further experimentation that might be done.

2 Experiments

2.1 Materials

Our eventual goal is to develop a tool for analyzing time-series expression data on *E. coli* as it is produced by the Blattner Lab at the University of Wisconsin. But to test our hypotheses, this paper reports the analysis of time-series gene expression data from [Khodursky, Peter, Cozzarelli, Botstein, Brown and Yanofsky 2000]. This data set is used because it is focused on tryptophan metabolism, a well studied regulatory process, so it is an excellent check for the reverse engineering of a genetic network. In addition the authors already have generously made the data publicly available on a web page supplement to their paper, so computational experiments with this data can be replicated if desired. It should be noted that a common problem with current microarray expression data is a small number of data points, and this is especially true of time-series data. The present data set consists of 8 data points, from 4 time steps under tryptophan-rich conditions and 4 time steps under tryptophan-starved conditions. Nevertheless it is hoped that discretization and reasonable priors will permit useful results to be obtained. We hope to have significantly larger time-series data sets in the near future.

We obtained the operon map of known operons of *E. coli* from [Salgado, Moreno-Hagelsieb, Smith and Collado-Vides 2000] and the operon map of predicted operons of *E. coli* from [Craven, Page, Shavlik, Bockhorst and Glasner 2000] to build our initial DBN structure.[1] An operon is a sequence of genes that are transcribed together into mRNA on their way to being expressed as proteins. The presence or absence of these proteins, as well as other molecule(s) such as the amino acid tryptophan, may cause other genes to increase or decrease in transcription levels.

There are two important reasons to incorporate explicit operon nodes into the DBN model even though operon transcription levels are not observed. First, if we use nodes for genes only, and allow the learning algorithm to induce arcs between genes, it will induce many "useless" arcs between genes in the same operon. For example, if gene1 and gene2 are both in operon1, then we would expect the expression level of gene1 to be an excellent predictor of the expression level of gene2, but this would provide no new insight given that we already know (or believe) that gene1 and gene2 are in the same operon. Second, incorporation of operons in the model can help combat problems due to noise. For example, the trp operon (also known as trpEDCBA) contains five genes: trpA, trpB, trpC, trpD, and trpE. Because of noise in microarray experiments, the measured expression level for trpC might be a bit too high. But the five different gene expression measurements give us essentially five independent indicators of trp transcription, potentially reducing the effect of noise in the measurement of trpC expression.

The Bayes Net Toolbox software package written by [Murphy 2000] was used for the experiments in this paper because it already provided the necessary func-

[1] The full operon map, with an interactive graphical interface, will be available online within a month.

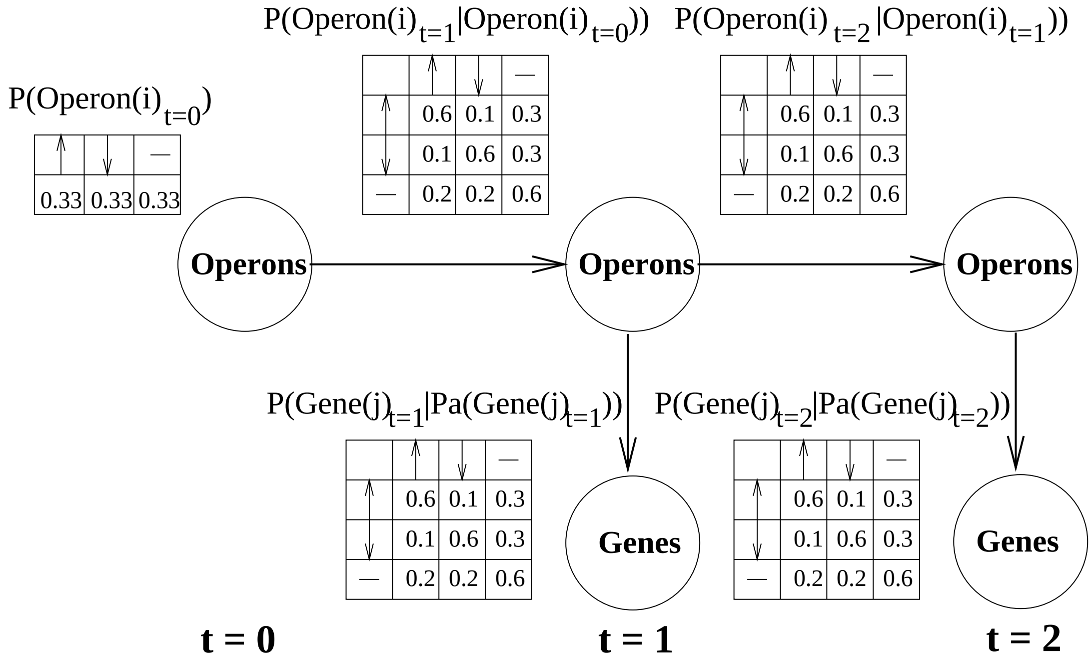

Figure 1: High-level Dynamic Bayes Net structure with CPTs for each arc in the model. $\uparrow$ represents up regulation $\downarrow$ represents down regulation and $-$ represents no change in expression level. Time slices are represented by t=0, t=1 and t=2.

tionalities for building Bayes Nets, as well as an implementation of EM for learning CPTs. We constructed the initial Bayes Net structure and learned the parameters of the model using the methods provided in Bayes Net Toolbox. Within this framework we implemented the structure search described in the next subsection.

2.2 Methodology

We start by describing our initial DBN structure. Each time slice in a DBN is identical in structure to the next, so we will first focus on the structure within a time slice. By definition, an operon is a cluster of contiguous genes that are transcribed together. Since an operon's transcription level affects the expression levels of the genes in that operon, we show this causality with arcs from each operon to its associated genes. Because we cannot directly measure an operon's expression levels, we represent each operon as a hidden node in the network. The gene expression levels, which can be measured, are our observed variables. The high-level structure of the initial DBN model is shown in Figure 1.

The arcs connecting operons to genes are known from our operon map. If operon1 consists of gene2 and gene3, our DBN will contain an arc from the operon1 node to the gene2 and gene3 nodes. This leaves undetermined the arcs from the hidden variables of one time step to the hidden variables of the next time step. Since an operon's expression level from one time step typically affects its expression level at the next time step, we add these arcs as shown in the detailed DBN structure in Figure 2. Any additional arc among hidden nodes, as well as all posterior CPT probabilities, must be inferred from time-series microarray data for *E. coli*.

The evidence variables in our DBN are the discretized gene expression levels from the experiments with excess tryptophan and tryptophan starvation. We define up regulated ($\uparrow$), down regulated ($\downarrow$) or no change($-$) as the possible discrete values. In particular, we compare the expression levels between two consecutive time-series measurements to determine whether there was a 1.4-fold increase ($\uparrow$), 1.4-fold decrease ($\downarrow$), or no change ($-$). Note that we are determining the relative change in expression from one time-step to another rather than absolute absent, present or marginal calls. We chose to use 1.4-fold difference as the threshold for determining change in expression levels because 1.4-fold is the smallest average extent of repression according to [Khodursky, Peter, Cozzarelli, Botstein, Brown and Yanofsky 2000]. Our best guess of informative priors for setting initial CPT values are shown in Figure 1.

Because of limited data, we consider only simple structural models in which each operon has at most two incoming arcs, from (1) the same operon at the previous time step, and (2) at most one other operon from the previous time step. Section 3 discusses the potential for relaxing this requirement. We employ broadly the structural EM methodology of [Friedman 1998]. Each operon begins with one parent—the same operon at the previous time step. In our full algorithm, for each operon we consider adding a different operon from the previous time step as a second parent. Each potential parent is considered. For each such potential second parent, the EM algorithm is employed to update all CPTs in the model to give a (local) maximum log likelihood. If any choice of second parent increases the log likelihood, then the choice that provides the highest log likelihood is selected.

In general, the preceding cycle through all the operons may need to be repeated several times for convergence to a locally optimal structure. Unfortunately, even with our structural restrictions, because we are using 142 operons and 169 genes the EM algorithm as implemented in the BN Toolbox for Matlab requires more than 10 minutes real time on a standard workstation (Sun running Solaris or Dell running Red Hat Linux). Even one full cycle requires 20022 calls to the EM algorithm, which will require over 4 months to run. For the long-term, we are reimplementing the algorithm in C to run in parallel on a Condor [Litzkow, Livny and Mutka 1988] pool of networked workstations. For the short-term, we focus the algorithm on nine operons containing genes known to be involved in tryptophan metabolism; the algorithm cycles once through these only, but *all* 142 operons containing genes in the data set are considered as potential parents. For each operon we record both the best and second-best choices for the additional parent.

2.3 Results

The results of the experiments were mixed and are summarized in Table 1. It is disappointing that for only one of the nine operons known to be involved in tryptophan metabolism (aroL) was another of the nine chosen as the best additional parent (aroP). The probability of inducing at least one such arc simply due to chance is quite high, at 0.41.[2] Hence this result provides no support for our hypotheses. More encouraging is the emergence of the tryptophan operon itself (trp, also known as trpEDCBA) as the second best choice of additional parent for four of the nine operons, including the important tryptophan repressor (trpR), as well as the appearance of aroP as the second best additional parent for tnaAB. The probability of having this many of the nine tryptophan-related operons chosen as the best or second-best parent for another of the nine due to chance alone is only 0.01. While further experimentation is required, these results provide some initial evidence in favor of our first hypothesis. Figures 3-8 show the induced CPTs for the 6 cases where a tryptophan-related operon was chosen as the second parent for another tryptophan-related operon.

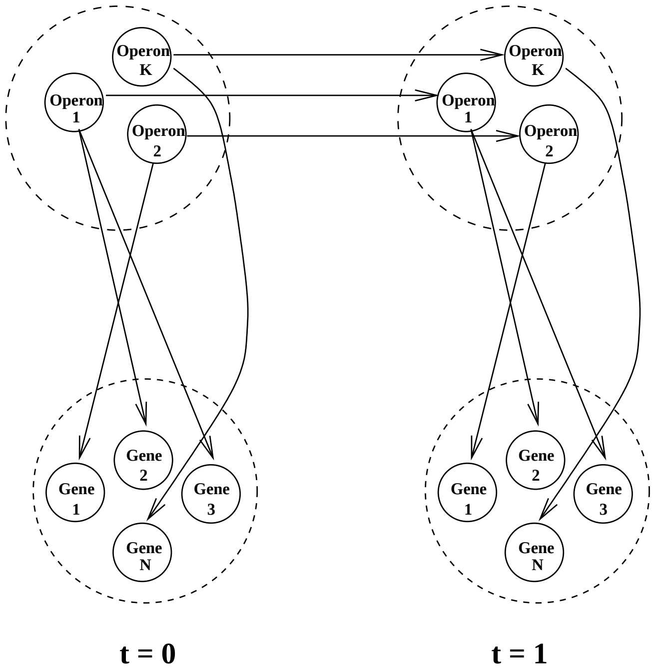

Figure 2: Dynamic Bayes Net structure details showing intra and inter time slice connections

It is interesting to note the significant repetition among the other best or second-best parents that are not members of the set of nine known tryptophan-related operons. The operon flgBCDEFGHIJK appears four times as a best or second-best parent. The predicted (from [Craven, Page, Shavlik, Bockhorst and Glasner 2000]) operons yafDE, gltJKL, and yciGFE appear two times each. These results suggest that perhaps these operons also play some role in tryptophan metabolism. [Khodursky, Peter, Cozzarelli, Botstein, Brown and Yanofsky 2000] note that in their cluster analysis yciGF forms a tight cluster with trpR and related operons and hence merits a closer look.

Somewhat surprisingly the tryptophan repressor (trpR), which plays a major role in the regulation of tryptophan metabolism, does not emerge as a best or second-best additional parent for any of the nine

[2] That is to say, the probability of at least one of the nine operons relevant to tryptophan metabolism being chosen as a parent for one of the other nine due to chance alone is 0.41. This figure is just the probability of one or more successes drawn from the binomial distribution $b(9, \frac{8}{141})$, since we draw a parent for each one of the nine tryptophan-related operons, and the probability of this parent being another one of the nine is $\frac{8}{141}$

Table 1: Most Probable DBN Structure. Operons known to be involved in the tryptophan metabolism regulatory pathway are in bold. Other operons that appear as parents more than once are in italics.

Operon Name	Most-probable parent	Second-most-probable parent
aroF	*gltJKL*	**trp**
aroG	*yafDE*	*flgBCDEFGHIJK*
aroH	*gltJKL*	**trp**
aroL	**aroP**	*flgBCDEFGHIJK*
aroP	*yciGFE*	*flgBCDEFGHIJK*
mtr	*yciGFE*	**trp**
tnaAB	*yafDE*	**aroP**
trp	ykfE	argCBH
trpR	*flgBCDEFGHIJK*	**trp**

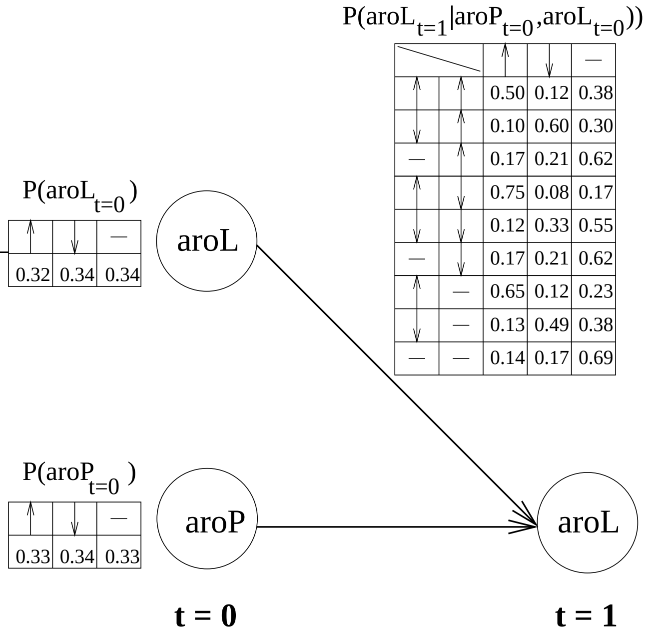

Figure 3: CPT for aroL conditional on aroL and aroP from previous time step.

tryptophan-related operons. Nevertheless, the tryptophan operon (trp) is the second-best additional parent for trpR. In the regulatory pathway for tryptophan metabolism, expression of trpR directly affects trp expression, but the influence of trp on trpR is indirect. Hence while the DBN picked up the link between trp and trpR, it did not correctly identify the direction of causality. In fact, in every case above where a tryptophan-related operon was chosen as the best or second-best parent of another tryptophan-related operon, the relationship between the operons in the regulatory pathway either flows in the opposite direction or is a relationship of indirect influence rather than direct. Thus the experiments refute our second hypothesis, that DBN structure learning will induce arcs that *actually appear* in the regulatory pathway.

3 Conclusions and Future Directions

We have reported an initial experiment in learning Dynamic Bayesian Networks as a means of modeling time-series gene expression microarray data, with the aim of gaining insights into regulatory pathways. The prior structure and prior CPTs of our DBN encode background knowledge about gene expression in the organism being modeled, *E. coli*. The experiment provides evidence that DBN learning is capable of identifying operons in *E. coli* that are in a common regulatory pathway. But it also provides evidence against the hypothesis that DBN learning is capable inducing arcs that reflect causality, or arcs that actually appear in the pathway. Nevertheless, this initial foray into DBNs for time-series data has several shortcomings that immediately suggest directions for further research.

First, use of a larger data set may improve the performance of the approach. Some additional time-series data recently have been made available by the Blattner Laboratory at the University of Wisconsin, under a different set of conditions, and we anticipate the availability of further time series data on *E. coli* in the year ahead. Nevertheless, potentially offsetting any such gain is the need to include additional genes (observed variables) and operons (hidden variables) in the analysis. The present data set used only 169 genes appearing in 142 operons. But the full *E. coli* genome has over 4000 genes, and the predicted operon map has well over 1000 operons.

A second, and perhaps more important, shortcoming of the present work is that computation time did not permit our full algorithm to be employed. The full algorithm modifies incoming arcs to every hidden node. As the arcs coming into one hidden node are modified, and the CPTs updated, this node may become a better parent for another node. A cascade of such improvements could dramatically improve the fit of the model and hence, potentially, the match of the model with the actual regulatory structure of the organism. Therefore, a crucial direction for further work is to increase the efficiency of our implementation of the learning algorithm, both through parallel execution on a Condor pool and simple reimplementation in C, so that the full algorithm can be tested. The faster implementation also will facilitate more extensive experimentation, including cross-validation to estimate the accuracies of expression levels that the model predicts

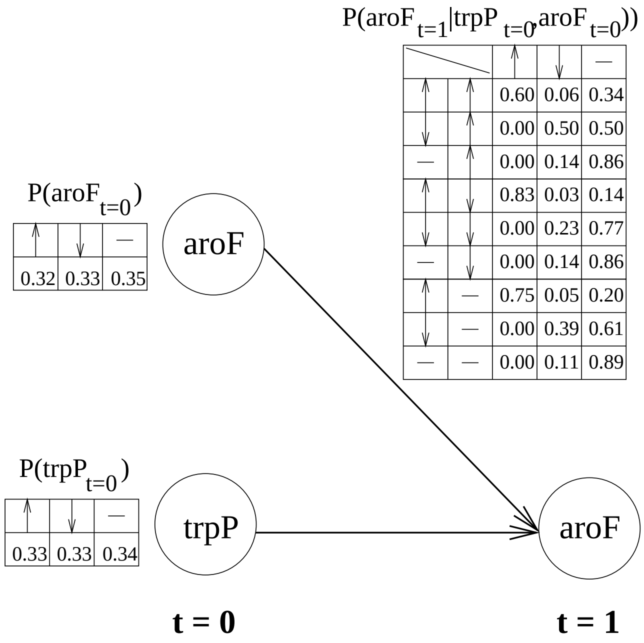

Figure 4: CPT for aroF conditional on aroF and trp from previous time step.

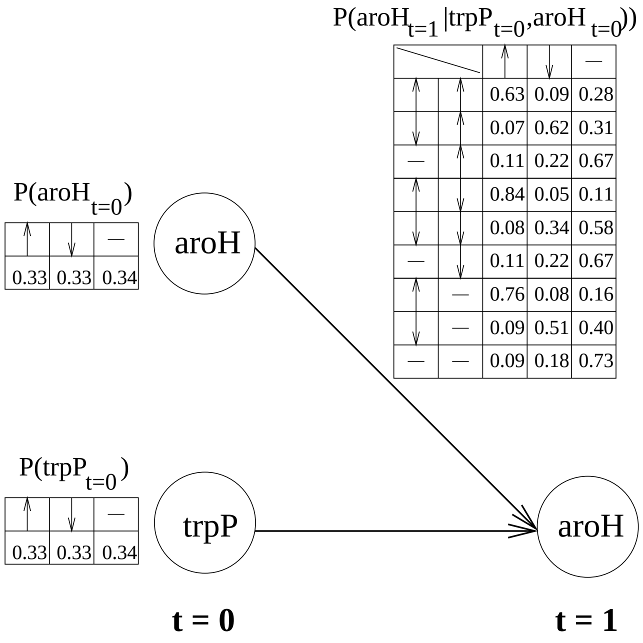

Figure 5: CPT for aroH conditional on aroH and trp from previous time step.

for various genes at various time steps.

Third, our initial DBN structure and prior CPTs are based on several simplifications. We ignore *attenuation* of operons—cases where at times only the first several genes in an operon are transcribed. The CPTs for genes based on their operons should reflect current knowledge about attenuation. For example, consider an operon for which attenuation is known to be important. If the transcription of the operon increases then perhaps earlier genes in the operon should have a higher probability of increased expression than should later genes in the operon. Another simplification is that we do not directly model important environmental factors. The model may perform better if we include additional hidden or observed variables corresponding to temperature or the availability of resources such as glucose or tryptophan.

This paper has presented the first application (to our knowledge) of Dynamic Bayesian Networks to time-series gene expression microarray data. It also has shown how background knowledge about an organism's genome (in this case, an operon map) can be used to construct the initial, core structure of the DBN. This background knowledge can be taken from the scientific literature or can itself be the output of another modeling system. In this case, the operon map consisted partially of each type of knowledge. The paper has provided some evidence that the results of such an application of DBNs provide additional insights into the organism's regulatory network. The paper also has demonstrated, though, that our DBN approach is much less useful for inducing direct causal links, that is, direct arcs in the regulatory network. Further experiments will provide insight into whether this shortcoming is inherent to DBNs or is merely a result of limited data and computational resources, as well as a result of our simplifying assumptions as described in the previous paragraph.

Acknowledgements

The first author was supported by NIH Biotechnology Training Grant NIH 5 T32 GM08349. The second author was supported by NSF Grant 9987841.

References

M. Craven, D. Page, J. Shavlik, J. Bockhorst and J. Glasner (2000). A Probabilistic Learning Approach to Whole-Genome Operon Prediction. *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, 116-127. La Jolla, CA. AAAI Press.

N. Friedman (1998). The Bayesian Structural EM Algorithm. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 129-138. Madison, WI:Morgan Kaufmann.

N. Friedman, M. Linial, I. Nachman and D. Pe'er (2000). Using Bayesian Networks to Analyze Ex-

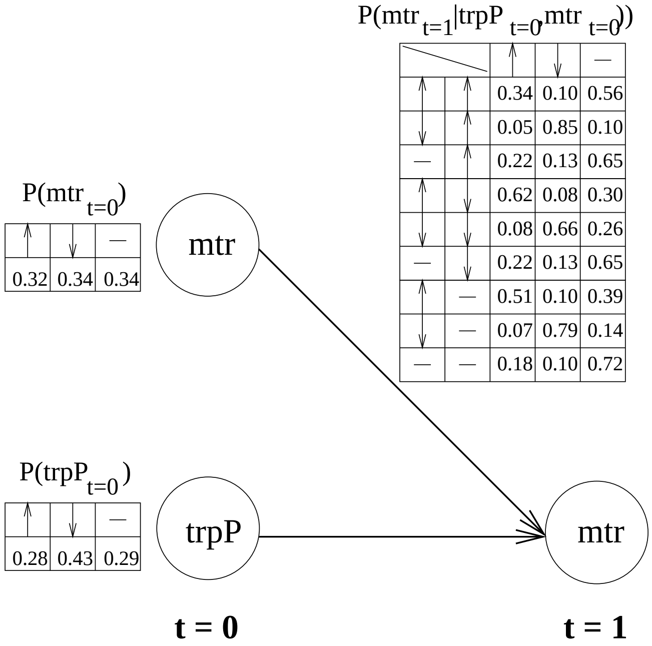

Figure 6: CPT for mtr conditional on mtr and trp from previous time step.

pression Data. *Journal of Computational Biology*, **7**(3/4):601-620.

A. Khodursky, B. J. Peter, N. R. Cozzarelli, D. Botstein, P. O. Brown and C. Yanofsky (2000). DNA microarray analysis of gene expression in response to physiological and genetic changes that affect tryptophan metabolism in *Escherichia coli. Proceedings of the National Academy of Science* USA **97**(22):12170-12175.

M. J. Litzkow, M. Livny and M. W. Mutka (1988). Condor: A hunter of idle workstations. *Proceedings of the Eighth International Conference on Distributed Computing Systems* 104-111.

K. Murphy and S. Mian (1999). Modeling Gene Expression Data using Dynamic Bayesian Networks. *Technical Report, Computer Science Division, University of California, Berkeley, CA*

K. Murphy (2000). Bayes Net Toolbox. http://www.cs.berkeley.edu/˜murphyk/Bayes/bnt.html.

H. Salgado, G. Moreno-Hagelsieb, T. F. Smith and J. Collado-Vides (2000). Operons in Escherichia coli: genomic analyses and predictions. *Proceedings of the National Academy of Science* USA **97**(12):6652-6657.

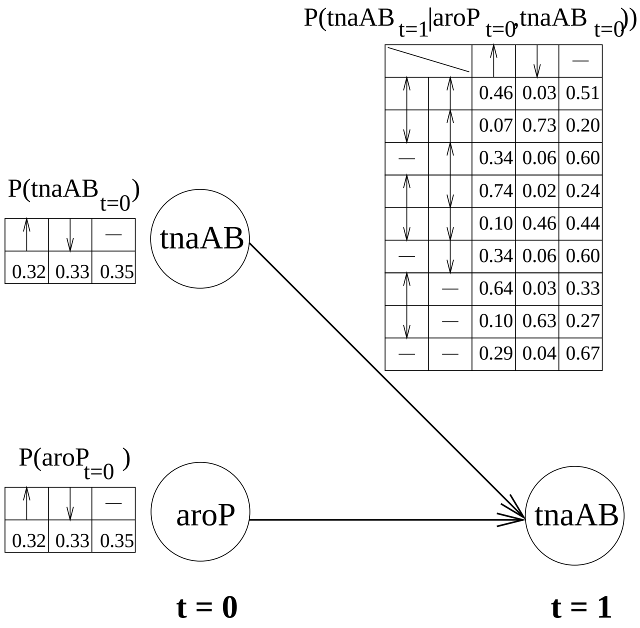

Figure 7: CPT for tnaAB conditional on tnaAB and aroP from previous time step.

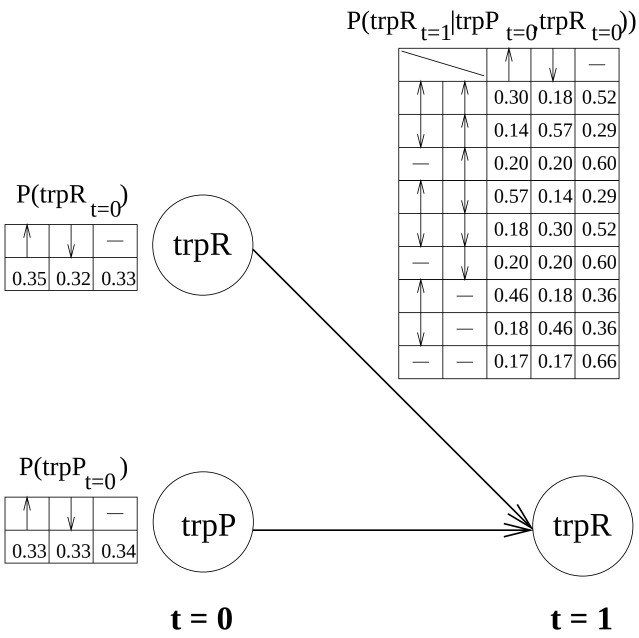

Figure 8: CPT for trpR conditional on trpR and trp from previous time step.