

# 1 Building Gene Networks with Time-Delayed Regulations

2 Iti Chaturvedi, Jagath C. Rajapakse

---

## 3 Abstract

4 We propose a method to build gene regulatory networks (GRN) capable of  
5 representing time-delayed regulations. The gene expression data is repre-  
6 sented in a linear model using a dynamic Bayesian network (DBN) and a  
7 skip-chain model using a hidden Markov model. The method therefore finds  
8 both short- and long-term regulatory interactions. The algorithm was tested  
9 on time-series data of the yeast cell-cycle. We compare the accuracy of GRN  
10 built by the present method with those built by using a higher-order DBN.  
11 The proposed method better fits the expression data and found core genes  
12 that are crucial in cell-cycle regulation.

13 *Key words:* Dynamic Bayesian networks, Gene regulatory networks,  
14 Viterbi algorithm, Skip-chain model, Genetic algorithms

---

## 15 1. Introduction

16 Gene expressions if collected over enough time points can be used to derive  
17 gene regulatory networks(GRN)(1). The GRNs represent causal activities  
18 of genes and gene products in biological systems and provide a basis for  
19 signal transduction in biological pathways. Since the signal transduction is  
20 transient, the study of the dynamics of the transduction is essential. The  
21 existing methods of deriving GRN from gene expression time-series can be

22 broadly classified into three categories: networks built by using 1. boolean  
23 rules 2; differential equations; and 3. stochastic modeling (2). Boolean  
24 networks are known to have lower sensitivity than Bayesian networks (3).  
25 They are not causal and use mutual information and minimum regulation  
26 linkage. Ordinary differential equations (ODE) have very high complexity as  
27 they describe processes at a very refined level (4).

28 Bayesian networks (BN) have been introduced for building gene regula-  
29 tory networks in the stochastic framework. Pathways have a natural repre-  
30 sentation in BN where genes are present at the nodes of the network and  
31 the edges represent causal interactions among them. The causal dependen-  
32 cies are in terms of conditional probabilities which infer 'cause and effect'  
33 relationships among the genes in the network. However, BN are acyclic, and  
34 cannot track time-delayed, feedback, and self-regulatory events. Building  
35 gene regulatory networks has been extended by using the dynamic Bayesian  
36 networks (DBN) where the parents are selected from the previous time in-  
37 stant and the time-series are assumed to be first-order stationary (5). The  
38 first-order assumptions allow feedbacks but still deprive DBN of representing  
39 variable time-delayed interactions. The DBN formulation of GRN has been  
40 extended to higher-order which are capable of extracting higher-order regu-  
41 latory interactions. Mutual information has been used to determine the best  
42 time-delay (6). These generative models become intractable in very high-  
43 orders. Therefore, we resort to a conditional skip-chain model which take  
44 into account delayed regulatory events of different orders.

45 We use a skip-chain model where two types of features model the time-  
46 delayed regulations: 1. linear features modeling the lower-order delays; and

47 2. skip features modeling the long-distant delays. In our model, the skip-  
48 features are modeled by using a hidden Markov model (HMM) where the  
49 log likelihood of the network is decomposed into a sum of consecutive pairs  
50 of genes, so the maximum likelihood regulation can be found by using the  
51 Viterbi algorithm. The Viterbi skip-features automatically determine the  
52 best time delay in the higher-order Markov chain.

53 Our approach consists of two stages: 1) identification of time-delayed  
54 interaction features and computation of Viterbi scores; and 2) prediction of  
55 the optimal GRN by using a genetic algorithm (GA). The fitness function of  
56 the genetic algorithm includes the Viterbi scores of time-delayed interactions.  
57 We demonstrate our method with an application to a long time-series of yeast  
58 cell-cycle data. Our method finds core genes that have regulatory effects with  
59 different time-delays on the cell cycle.

60 Earlier fusion of PPIN and GRN has been attempted using a similar  
61 model (7). The work was extended in (8) to implement a GRN without prior  
62 PPIN. The paper also considered over-fitting problems for small datasets.  
63 In this paper a derivation of the skip-chain model is given. We consider  
64 larger sets of genes and higher number of time points. Discretization into 3  
65 levels and effect of multiple runs of GA is explored. Lastly, validation has  
66 been done with Biogrid. A validation with Biogrid PPI (9) for higher-order  
67 interactions shows that the method is more effective than simple HDBN.

## 68 2. Methods

69 Consider a set of  $n$  genes  $G = \{g_i : i = 1, 2, \dots, n\}$  and time-series of  
70 gene expressions gathered over  $T$  time points for all the genes. Let the

71 gene expression data be  $x = \{x_{i,t}\}_{n \times T}$  in which row vector  $x_i = (x_{i,t} : t =$   
72  $1, 2, \dots, T)$  corresponds to the gene expression time-series of gene  $g_i$ . The  
73  $x_{i,t}$  denotes the expression level of gene  $g_i$  at time  $t$ . Suppose that gene  
74 expressions are discretized into a set  $\Gamma$  of  $d$  levels:  $\Gamma = \{1, 2, \dots, d\}$ . A level  
75 of gene expression indicates a state of the gene. Let the set of parent genes  
76 regulating the gene  $g_i$  be denoted as  $a_i$  and the number of states that a node  
77 in  $a_i$  take to be  $q_i$ .

### 78 2.1. Bayesian Networks (BN)

79 BN are causal networks that can represent regulations among the genes  
80 at the nodes, as edges in the network. The BN decomposes the likelihood  
81 of gene expressions into a product of conditional probabilities by assuming  
82 independence of non-descendant genes, given their parents :

$$p(x) = \prod_{i=1}^n p(x_i | a_i, \theta_i) \quad (1)$$

83 where  $x = (x_1, x_2, \dots, x_n)$ ,  $p(x_i | a_i, \theta_i)$  is the conditional probability of gene  
84 expression  $x_i$  given its parents  $a_i$ , and  $\theta_i$  denotes the parameters of the con-  
85 ditional probabilities.

86 Given the set of conditional distributions with parameters  $\theta = \{\theta_i : i =$   
87  $1, 2, \dots, n\}$ , the likelihood can be written as

$$p(x) = \int p(x|S, \theta) p(\theta|S) d\theta \quad (2)$$

88 Let  $\theta_{ijk} = P(x_{i,t} = k | a_i = j)$  and  $N_{ijk}$  be the number of instances of  $\theta_{ijk}$  that  
89 occur in the training data. Using the property of decomposability (5),

$$P(x) = \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^d \theta_{ijk}^{N_{ijk}} \quad (3)$$

90 The model parameters  $\theta_{ijk}$  are estimated by using maximum likelihood (ML):

$$\theta_{ijk} = \frac{N_{ijk}}{\sum_{k=1}^d N_{ijk}} \quad (4)$$

91 Then the log-likelihood of the data is given by

$$\log P(x) = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^d N_{ijk} \log \frac{N_{ijk}}{\sum_{k=1}^d N_{ijk}} \quad (5)$$

92 The likelihood approximation is known to be good when a large amount of  
 93 data points are available (10) . Therefore, we use the maximum likelihood  
 94 estimate of parameters to obtain the optimal structure of the Bayesian net-  
 95 work.

## 96 2.2. Dynamic Bayesian Networks (DBN)

97 The acyclic condition of BN does not allow self- and feedback-regulations  
 98 of genes, which are essential characteristics of GRN. The dynamic Bayesian  
 99 networks (DBN) overcome this by modeling the regulatory network from one  
 100 time point to the next and the temporal expression pattern by unrolling  
 101 network over time. A first-order DBN is defined by a transition network  
 102 of interactions between a pair of structures  $(S_t, S_{t+1})$  corresponding to time  
 103 instances  $t$  and  $t + 1$ . In time instance  $t + 1$ , the parents of genes are those  
 104 specified in the time instant  $t$ . The gene regulations are obtained by un-  
 105 rolling the transition network over time and assuming first-order stationary  
 106 behaviour over time. The likelihood of the data is given by Eq. (3):

$$P(x) = \prod_{t=1}^T \prod_{i=1}^n \prod_{j=1}^{q_i} \prod_{k=1}^d \theta_{ijk}^{N_{ijk}^{(t,t+1)}} \quad (6)$$

107 where  $N_{ijk}^{(t,t+1)}$  correspond to the number of instances where  $x_{i,t+1} = k$  while  
 108  $a_{i,t} = j$ . The first-order DBN has two layers and therefore  $2n$  nodes.

109 *2.3. Hidden Markov Models (HMM)*

110 The classical DBN is unable to capture complex time-dependencies and  
 111 is extended to an  $o$ -order ( $o \geq 2$ ) Markov chain. It predicts the expression  
 112 levels of a set of genes based on the expressions of up to  $o$  previous time  
 113 points using frequency statistics. Higher-order dynamic Bayesian networks  
 114 (HDBN) have been proposed to study time-delayed interactions. However,  
 115 as  $o$  increases it is not possible to compute the statistics using the few time  
 116 points. Over-fitting occurs as can be seen by a corresponding decrease in  
 117 likelihood in higher orders.

118 Therefore, we resort to a first-order hidden Markov model (HMM) to de-  
 119 termine delayed interactions. It determines the probability of expression of  
 120 a gene  $g_j$  at time point  $t$ , given that  $g_i$  was observed at  $s_t$  where  $s_t < t - 1$   
 121 within the section of the time-series of length  $t - s_t$ . Let a sequence of hidden  
 122 states from time point  $s_t$  to  $t$  be denoted by  $y_{s_t:t} = (y_{s_t}, y_{s_t+1}, \dots, y_t)$  where  $y_{t'}$   
 123 denotes the gene expressed at time point  $t'$  in the path. Correspondingly, we  
 124 have the observed data  $x_{s_t:t} = (x_{i',s_t}, x_{i',s_t+1}, \dots, x_{i',t})$  where  $x_{i',t'} = k$  is the  
 125 discretized gene expression state. Since only the upregulation, downregula-  
 126 tion or no regulation is considered, we only consider  $x_{i',t'} \in -1, 0, 1$ .

127 Given the microarray data, the maximum likelihood estimation can be  
 128 used to estimate the state transition and emission probabilities, which are  
 129 defined as follows (11):

$$a_{l,m} = \frac{M_{l,m}}{\sum_{m'=1}^n M_{l,m'}}, \quad \forall y_{t'} = g_l, y_{t'+1} = g_m \in G \quad (7)$$

$$b_l(k) = \frac{M_l^k}{\sum_{k'=1}^d M_l^{k'}}, \quad \forall k \in \Gamma, t' \in \{s_t, s_t + 1, \dots, t\}, g_l \in G. \quad (8)$$

130 where  $M_{l,m}$  denotes the number of occurrences where  $x_{l,t'} = x_{m,t'+1} = 1$

131  $\forall t' \in \{s_t, s_t + 1, \dots, t\}$ ,  $\forall k \in \Gamma$  and  $M_l^k$  denotes the number of occurrences  
 132 where gene  $g_l$  has been at discrete state level  $k$ ,  $\forall t \in \{s_t, s_t + 1, \dots, t\}$ .

#### 133 2.4. Viterbi Algorithm

134 When the expression time-series are modeled with an HMM, the max-  
 135 imum a posteriori (MAP) estimate could be used to find the time-delayed  
 136 interactions of a pair of genes. The path begins and ends at the known states  
 137 of genes : say,  $y_{s_t} = g_i$  and  $y_t = g_j$ . We assume that  $t - s_t$  is not very large  
 138 and conditional independence between feature vectors. For a sequence of a  
 139 set of genes, the most probable path is given by the MAP estimate:

$$\arg \max_{y_{s_t:t}} p(y_{s_t:t}|x) = \arg \max_{y_{s_t:t}} p(x|y_{s_t:t})p(y_{s_t:t}) \quad (9)$$

140 The Viterbi algorithm(VA) can be used to find the best path by finding  
 141 the MAP estimate, between two genes at distant time points (12). The  
 142 state transition and emission probabilities are estimated during training. VA  
 143 is a dynamic programming procedure and determines the best path in an  
 144 incremental manner. Let  $\delta_m(t')$  be the probability of the most probable path  
 145 ending at gene  $g_m$  with the observation  $x_{m,t'}$  at time  $t'$ . Then, the best path  
 146 at the next iteration is found as

$$\delta_m(t' + 1) = b_m(k) \max_l \{\delta_l(t') a_{l,m}\} \quad (10)$$

147 We can divide the path probability by length of path to get a first-order  
 148 probability as a goodness of fit of the path. We define skip-edge score as the  
 149 normalized MAP interaction :

$$h(x_i, a_i, s_t, t) = \log \frac{1}{(t - s_t)} \max_{y_{s_t:t}} p(y_{s_t:t}|x) \quad (11)$$

150 where the parent set  $a_i$  has only one gene at time point  $s_t$ .

151 Finally, for any pair of genes, we can choose the best time-delayed inter-  
 152 action having the highest probability :

$$\hat{h}(x_i, a_i, s_t, t) = \max_{s_t, t} h(x_i, a_i, s_t, t) \forall t - s_t > o \quad (12)$$

153 where  $g_j \in a_i$  and  $o$  is predefined linear order. Next we combine this with  
 154 the HDBN using a skip-chain model.

### 155 2.5. Linear and Skip features

156 In order to handle both short- and long-range interactions, we model gene  
 157 regulations by using both linear- and skip-chain features. Linear and skip  
 158 features in microarray data are illustrated in Figure 1. The up-regulated  
 159 genes are indicated with one. The genes  $g_2$  and  $g_3$  have a linear second-order  
 160 interaction as both genes are upregulated at a delay of two time points.  
 161 Similarly, the order of interaction between  $g_1$  and  $g_2$  is four and is hence  
 162 represented by a skip feature. There can be numerous skip-features between  
 163 a pair of genes. These features could be of different time delays or have the  
 164 same time delay with different start and end time points. Here, we use our  
 165 method of choosing an optimal time-delayed skip-feature between two genes  
 166 as described in the previous section.

167 We can interpolate the two types of features by expressing the likelihood  
 168 of a gene expression  $x_i$  as a weighted sum of linear and skip-edge scores:

$$\log p(x_i | a_i, \theta_i) \propto \lambda f(x_i, a_{i(t-o:t)}, t) + (1 - \lambda) h(x_i, a_i, s_t, t) \quad (13)$$

where from (Eq. 5)  $f(x_i, a_{i(t-o:t)}, t) = \sum_{j=1}^{q_i} \sum_{k=1}^d N_{ijk} \log \frac{N_{ijk}}{\sum_{k=1}^d N_{ijk}}$

169 where  $f(x_i, a_{i(t-o:t)}, t)$  and  $h(x_i, a_i, s_t, t)$  represent the linear- and skip-feature  
170 functions and  $\lambda$  is a weight determined heuristically.

171 Linear-chain feature functions  $f(x_i, a_{i(t-o:t)}, t)$  represent local dependen-  
172 cies that are consistent with an  $o$ -order Markov assumption of gene expres-  
173 sions (13). The skip-chain features represent long range dependencies in a  
174 GRN (14). The skip-chain feature functions  $h(x_i, a_i, s_t, t)$  exploit the depen-  
175 dencies between genes that are arbitrarily distant at time instances  $s_t$  and  
176  $t$  respectively. Such a skip-feature models a variable length Markov chain  
177 up to  $T - 1$  order where  $T$  is number of time points. We use an HDBN  
178 to implement a linear-chain model and first-order HMM to implement the  
179 skip-chain model.

180 A classical HDBN uses a GA to find the optimal delays. For a DBN  
181 each gene given its parents needs  $d^{|a_i|} \times d$  parameters, where  $d$  is number of  
182 discrete levels and  $|a_i|$  is the cardinality of parent set. For  $o$ -order HDBN,  
183 we can further have  $o^{|a_i|}$  structural possibilities for each DBN. Hence, the  
184 search space and corresponding complexity is very high to find delays. On  
185 the other hand, skip models use VA to find the optimal delay and associated  
186 probability. Complexity of Viterbi is known to be quadratic on length of  
187 delay which is much smaller than the complexity of GA.

### 188 3. Implementation using a Genetic Algorithm

189 A genetic algorithm (GA) is used to find the optimal network structure.  
190 The individual solutions in the GA is defined by the connectivity matrix  
191  $\{c_{i,j}\}_{n \times n}$  where  $g_j$  is a parent of  $g_i$  in  $c_{i,j}$ . Each connection  $c_{i,j}$  is initial-  
192 ized from the values in  $M = \{M(i, j, l)\}_{n \times n \times T}$ , where  $M(i, j, l)$  is mutual

193 information between the expressions of genes  $g_i$  and  $g_j$  at a time lag  $l$ .

194 If each gene is allowed to have a maximum of  $N_p$  number of parents, then  
 195 the connections are randomly initialized as follows:

$$c_{i,j} = \begin{cases} \arg \max_l M(i, j, l) & \text{if } M(i, j, l) > \alpha \text{ and } \forall l \leq o \\ 0 & \text{if } M(i, j, l) < \alpha \text{ or } |a_i| > N_p \end{cases} \quad (14)$$

196 where  $|a_i|$  is number of parents of gene  $i$ ,  $\alpha$  is the threshold for mutual  
 197 information, and  $o$  is the order of the model. We randomize the order of  
 198 genes during initialization for each individual. The Bayesian score of this  
 199 graph of linear edges of low orders can be calculated using Eq. (5).

200 To account for longer delays, for any two genes  $g_i$  and  $g_j$  where  $c_{i,j} > 1$ , we  
 201 choose the highest Viterbi score among all the possible interaction features  
 202  $\hat{h}(x_i, a_i, s_t, t)$ . Here we only consider skip-edges longer than the linear edges  
 203 modeled using an HDBN. If there are no such skip-edges, we set the feature  
 204 probability to 0. Lastly we interpolate the probabilities of the linear and  
 205 skip-graph using the weight  $\lambda$  as in Eq. (14).

206 A random interpolation weight  $\lambda < 1$  can be appended to the individual.  
 207 The GA then finds the best structure with the highest posterior probabil-  
 208 ity for different combinations of linear score, skip score and  $\lambda$ . GA does  
 209 optimization of search for the structure, it parallel processes populations.  
 210 Mutation and crossover introduce changes in the structure. Here we run the  
 211 GA for  $Q$  generations or if the change in score is less than  $\alpha_q$  for 20 consecu-  
 212 tive generations. As the low lying structures can easily dominate the others  
 213 leading to premature search convergence, a minimum similarity threshold  
 214 of  $p_s > 0.7$  is maintained in each generation. Crossover involves swapping  
 215 several rows and the weights between two parents resulting in possibly lower

216 energy structures. Lastly the mutation operator in each generation selects a  
217 random individual and inverts a random interaction.

218 The gene expressions were normalized by linear transform, and each pro-  
219 file was divided by mean for the gene. A GA was used to determine  $T$  scaling  
220 factors. Spearman correlation is calculated between scaled expression and  
221 median profile for all genes. Fitness function of the GA is the sum of mean  
222 and variance of the correlation. Lastly, the scaled data is discretized into  
223 three levels based on relative increase between two consecutive time points.

#### 224 4. Experiments and Results

225 We evaluated our method on time-series gene expressions of yeast cell-  
226 cycle data obtained from Chou et al. (15) (17 time points) and Spellman  
227 et al (16) (24 time points, *cdc-15* cell cycle arrest). The yeast cell-division  
228 cycle consists of four main phases: genome duplication (S phase), and nuclear  
229 division (M phase), separated by two gap phases (G1 and G2). The S-G1-M-  
230 G2-S form a cycle for cell duplication. The expression values were normalized  
231 and discretized into 1 for upregulation, 0 for no regulation, and -1 to denote  
232 downregulation by using an approach described earlier (17). We use Chou  
233 dataset on nine genes which appear to control the sequential activation of  
234 cyclins and other cell cycle regulators (6). Similarly the Spellman *et al.*,  
235 dataset was used on subset of genes in different phases.

236 VA was used to compute skip-feature probabilities of all pairs of genes  
237 and GA was used to find the optimal structure. Simulation was done upto  
238 order four of HDBN and skip-chain, with a maximum skip-edge length of  
239 10 time points. We plotted the histogram of MI for each pair of genes for

240 different time delays. The peak of the histogram was taken as the threshold  
 241 as most interactions below that had negative or low mutual information. The  
 242 parameters of the GA were determined empirically. A MI threshold of 0.27  
 243 was found optimal to recover edges in Spellman dataset and 0.1 for the Chou  
 244 dataset. The skip-edge weight is determined heuristically by GA along with  
 245 structure. The GA chooses the network with the best combination of the  
 246 skip and linear edges Eq. (14). Simulation was done at different numbers of  
 247 individuals ( $N$ ) and generations ( $Q$ ) ( $N=200/300/400$  and  $Q=300/400/500$ )  
 248 for both HDBN and skip-chain model. The GA similarity threshold was set  
 249 to 0.7, and in the case of Chou dataset it was increased to 0.9 to avoid early  
 250 convergence. The GA stops when the maximum number of generations is  
 251 reached or if the score difference is 1 for 20 consecutive generations. We  
 252 consider edges with confidence over 0.7 over 20 runs of the GA in the final  
 253 network. The mean and standard deviations were reported. It is observed  
 254 that optimal  $\lambda$  found by GA is larger for small networks where probability of  
 255 skip-edge is low and is smaller for large networks where probability of skip  
 256 interactions becomes higher due to longer cascades of genes.

257 As seen from Table 1, HDBN of order four and skip-chain of order 1  
 258 have the highest likelihood in all datasets confirming that the network fits  
 259 expression data well. As seen, a skip-chain yielded higher number of edges at  
 260 0.7 confidence since the networks are more stable. The HDBN shows a peak  
 261 of the interactions at delay 1 and 4. This indicates that most interactions  
 262 are first-order or instantaneous, and the fourth-order is insufficient to capture  
 263 all higher-order interactions. Since, we are modeling variable order delay, a  
 264 previously proposed validation can be used (7). Here we look for a cascade

265 of genes in the GRN corresponding to an interaction in PPIN. On a subset  
266 of 19 S phase genes for which interaction are available in Biogrid we can  
267 clearly see that our model gives higher number of true positives than DBN  
268 or HDBN. Figure 2 shows predicted network for 9 genes, it can be seen that  
269 delays are consistent with phases of cell cycle. For eg. Ndd1 regulates Swi6  
270 at a delay of 70 mins. Figure 3 shows prediction for subset of 19 genes. The  
271 dashes edges correspond to cascade of genes forming a TP in PPI. For eg.  
272 HTB2 interaction with MET6, takes the form: HTB2 interacts with HTB1,  
273 and HTB1 interacts with MET6.

274 We also see that the method is robust to the increase in the number  
275 of genes. It can be seen that majority of the predicted interactions have  
276 time delays. Bigger networks like S(36) of 52 interactions had several 8  
277 time points delay. This is very useful as building higher-order DBN is very  
278 time-consuming and has to deal with an exponential number of parameters.  
279 Hubs in a network are nodes with high degree of connectivity, which usually  
280 represent important nodes in causal networks. Table 2 gives a list of top  
281 10 hubs of networks derived by different methods for 19 genes in S phase.  
282 The corresponding hubs in the Biogrid target network are also given. The  
283 top core genes produced by all methods seem the same and the core genes  
284 produced by the DBN and our method were quite similar.

285 Further comparison of top 10 hubs predicted by a DBN, an HDBN and a  
286 skip-chain using Saccharomyces Genome Database (SGD) showed that while  
287 a DBN had hubs involved in instantaneous events such as initialization, si-  
288 lencing, etc., the time-delayed hubs in HDBN were mostly regulatory or  
289 feedback associated. Our skip-chain model showed a combination of both

290 types of regulation, hence representing the robust signaling networks well.  
291 As seen in the S phase, some hubs such as HHH1, HTB1, HTA2, and HHT1  
292 are conserved in all the models. These are all histones required to initiate  
293 duplication by chromatin assembly and chromosome function. HDBN model  
294 picked up a hub ADA2 also seen in Biogrid. ADA2 encodes a chromatin mod-  
295 ifying complex. It also plays a role in transcriptional silencing at telomeres  
296 which occurs at end of duplication. KIP1 was a FP, it is a kinesin-related  
297 motor protein required for mitotic spindle assembly and chromosome seg-  
298 regation. It encodes the inhibitor of several cyclin CDK complexes which  
299 control the progression of the cell cycle from G1 to S phase. These events  
300 are slower and hence emerge in skip-chain model.

## 301 5. Discussion and Conclusion

302 Pathways are often triggered by transcription factors which in turn ex-  
303 press genes and produce proteins. Therefore, the regulatory interactions in  
304 molecular pathways can be given by GRN. Gene regulations generally in-  
305 cludes dynamic feedback loops, cascaded interactions, intermediary factors,  
306 etc., which provides for underlying biological mechanisms of regulation. This  
307 results in different time delays in regulatory interactions. The delays in reg-  
308 ulations are an integral part of biology. In this work we focused on modeling  
309 delays in GRN.

310 We have considered higher-order DBN (HDBN) for representing delays in  
311 regulations. When larger delays are involved, implementation of HDBN be-  
312 comes intractable. Therefore, we proposed a skip-chain HDBN. This involved  
313 two components: linear model to represent short delays and skip model to

314 represent long delays. These two components may represent actions of ac-  
315 tivator and inhibitor involved in regulatory interactions. Our method was  
316 evaluated against earlier approaches, which shows our method better fits the  
317 gene expression data when GRN was built. In order to provide a more biologi-  
318 cal meaningful validation, we performed comparison with the protein-protein  
319 interaction data. That validation also showed superiority of our technique  
320 over other methods but because of the incompleteness of protein-protein in-  
321 teraction data sources, such comparisons results in large false positives.

322 Skip-chain models address the difficulties of a HDBN by easily incorpo-  
323 rating long time-delayed regulations. The skip-chain model is a first-order  
324 HMM and captures long-distance dependencies of input time-course gene  
325 expressions. This inference technique leads to lower total training time with-  
326 out loss in accuracy compared to HDBN. The forward Viterbi path through  
327 the trellis determines the best long-distant time delay and therefore auto-  
328 matically finds the best higher-order interactions between two genes. The  
329 method can be applied to different long- time series by suitably tuning the  
330 GA similarity measures.

## 331 References

- 332 [1] Wagner J, Stolovitzky G: **Stability and Time-Delay Modeling**  
333 **of Negative Feedback Loops.** *Proceedings of the IEEE* 2008,  
334 **96(8):1398–1410.**
- 335 [2] Gebert J, Motameny S, Faigle U, Forst CV, Schrader R: **Identifying**  
336 **Genes of Gene Regulatory Networks Using Formal Concept**  
337 **Analysis.** *Journal of Computational Biology* 2008, **15(2):185–194.**

- 338 [3] Li P, Zhang C, Perkins EJ, Gong P, Deng Y: **Comparison of prob-**  
339 **abilistic Boolean network and dynamic Bayesian network ap-**  
340 **proaches for inferring gene regulatory networks.** *BMC Bioinform-*  
341 *atics* 2007, **8**:S13–S20.
- 342 [4] Liu B, Thiagarajan P, Hsu D: **Probabilistic Approximations of Sig-**  
343 **nalng Pathway Dynamics.** In *Computational Methods in Systems*  
344 *Biology* 2009:251–265.
- 345 [5] Friedman N, Murphy K, Russell S: **Learning the Structure of Dy-**  
346 **namc Probabilistic Networks.** *Proceedings of the 14th Annual Con-*  
347 *ference on Uncertainty in Artificial Intelligence (UAI-98)* 1998, :139–14.
- 348 [6] Zhengzheng X, Dan W: **Modeling Multiple Time Units Delayed**  
349 **Gene Regulatory Network Using Dynamic Bayesian Network.**  
350 In *Data Mining Workshops, 2006 ICDM Workshops 2006. Sixth IEEE*  
351 *International Conference on* 2006:190–195.
- 352 [7] Chaturvedi I, Rajapakse J: **Fusion of Gene Regulatory and Pro-**  
353 **tein Interaction Networks Using Skip-Chain Models.** In *Pattern*  
354 *Recognition in Bioinformatics, Volume 5265 of Lecture Notes in Com-*  
355 *puter Science* 2008:214–224.
- 356 [8] Chaturvedi I, Rajapakse J: **Detecting robust time-delayed reg-**  
357 **ulation in Mycobacterium tuberculosis.** *BMC Genomics* 2009,  
358 **10**(Suppl 3):S28.
- 359 [9] Breitkreutz BJ, Stark C, Reguly T, Boucher L, Breitkreutz A, Livstone  
360 M, Oughtred R, Lackner DH, Bahler J, Wood V, Dolinski K, Tyers M:

- 361       **The BioGRID Interaction Database: 2008 update.** *Nucl. Acids*  
362       *Res.* 2008, **36**(suppl1):D637–640.
- 363 [10] Friedman N, Murphy K, Russell S: **Learning the Structure of Dy-**  
364       **namic Probabilistic Networks.** *Proceedings of the 14th Annual Con-*  
365       *ference on Uncertainty in Artificial Intelligence (UAI-98)* 1998, :139–14.
- 366 [11] Cappae O, Moulines E, Rydaen T: *Inference in hidden Markov models*  
367       2005.
- 368 [12] Hao T, Huang TS: **Improved Graphical Model for Audiovisual**  
369       **Object Tracking.** In *Multimedia and Expo, 2006 IEEE International*  
370       *Conference on* 2006:997–1000.
- 371 [13] Galley M: **A Skip-Chain Conditional Random Field for Rank-**  
372       **ing Meeting Utterances by Importance.** In *Proceedings of the*  
373       *2006 Conference on Empirical Methods in Natural Language Process-*  
374       *ing (EMNLP 2006)* 2006:364–372.
- 375 [14] Sutton C, McCallum A: **Collective Segmentation and Labeling of**  
376       **Distant Entities in Information Extraction.** In *Presented at ICML*  
377       *2004 Workshop on Statistical Relational Learning* 2004.
- 378 [15] Chou R, Campbell M, Winzeler E, Steinmetz L, Conway A, Wodicka  
379       L, Wolfsberg Ta: **A genome-wide transcriptional analysis of the**  
380       **mitotic cell cycle.** *Molecular Cell* 1998, **2**:65–73.
- 381 [16] Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB,  
382       Brown PO, Botstein D, Futcher B: **Comprehensive identification of**

383 cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae*  
 384 by microarray hybridization. *Mol Biol Cell* 1998, **9**(12):3273–97.

385 [17] Shmulevich I, Zhang W: **Binary analysis and optimization-based**  
 386 **normalization of gene expression data.** *Bioinformatics* 2002,  
 387 **18**(4):555–65.

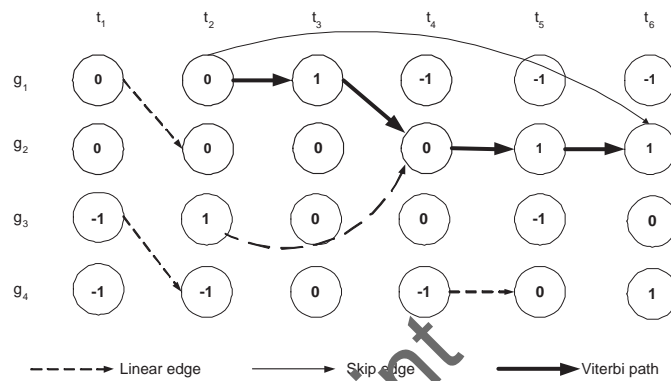


Figure 1: State transition diagram for six time points and four genes in a DBN. The dashed edges are linear  $o = 1, 2$  order edges from 1 by linear features. The solid directed edge is an example of skip-edge over four time points which models a long-distant dependency. The bold directed line shows a skip path computed by Viterbi algorithm.

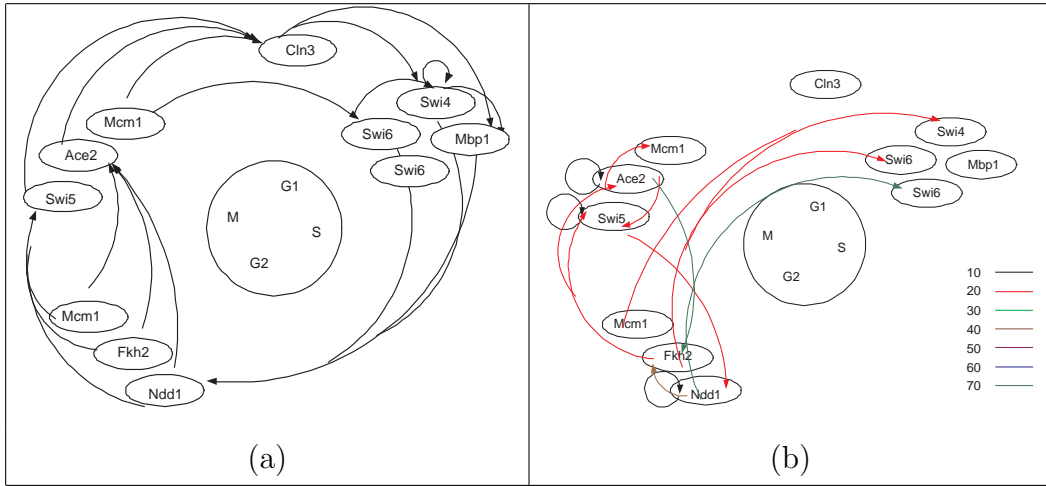


Figure 2: Time-delayed interactions in predicted network of 9 genes (a) Target network, (b) Predicted network

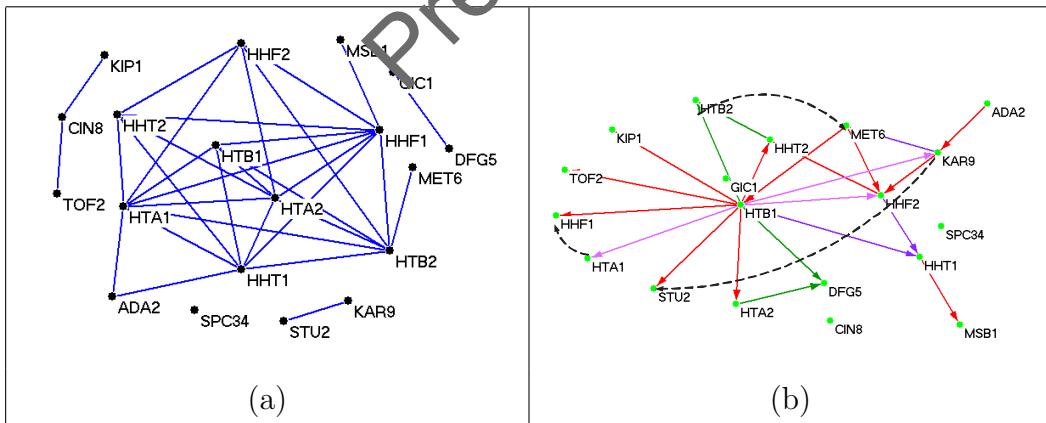


Figure 3: Time-delayed interactions in predicted network of 19 genes (a) Target network, (b) Predicted network. Dashed edges correspond to  $k$ -skip validation.

Table 1: First- and higher-order regulations in predicted by DBN(d), HDBN(h), and skip-chain(s) models, ( $n$ ) denotes overlapping skip-edges and  $o$  is order of the model.

#genes	$o$	ML	Order of Regulation								Tot	$k$ -TP
			1	2	3	4	5	6	7	8		
All(9)	d:1	$-52.66 \pm 3.13$	15								15	11
	h:4	$-34.95 \pm 0.53$	3	7	2	6					18	7
	s:1	$-29.95 \pm 0.77$	15	(9)		(1)		(2)			15	11
S(19)	d:1	$-194.57 \pm 13.34$	20								20	27
	h:4	$-120.25 \pm 7.76$	6	0	1	7					14	21
	s:1	$-53.36 \pm 0.80$	25	(12)			(3)	(4)	(3)		25	36
S(36)	d:1	$-426.75 \pm 2.76$	52								52	36
	h:4	$-245.91 \pm 12.65$	27	6	3						28	34
	s:1	$-131.27 \pm 2.22$	52	(32)	(2)	(2)	(2)	(7)		(3)	52	37
G2(33)	d:1	$-320.45 \pm 3.46$	34								34	10
	h:4	$-300.22 \pm 11.71$	12	10	10	12					44	9
	s:1	$-182.10 \pm 0.71$	50	(31)	(2)	(1)	(2)	(6)		(4)	50	13
M(60)	d:1	$-702.49 \pm 23.60$	49								62	19
	h:4	$-409.13 \pm 19.80$	15	12	15	35					67	22
	s:2	$-324.51 \pm 8.00$	33	27	(5)	(43)		(10)			64	18

Table 2: Top 10 hubs in different phases of yeast cell-cycle. The number below the gene name is the connectivity in the network.

	$o$	Rank of genes based on connectivity									
		1	2	3	4	5	6	7	8	9	10
S(19)	bg	HHF1	HTA1	HHT1	HTB2	HTA2	HTB1	HHT2	HHF2	ADA2	CIN8
		8	8	8	7	7	7	5	5	5	3
	d:1	HTB1	HHF2	HTA1	HHT2	HHT1	MET6	HTB2	KIP1	MSB1	TOF2
		10	5	4	3	2	2	2	2	2	2
	h:1	TOF2	HTB1	HHT2	HHF2	HHF1	HTA1	ADA2	-	-	-
8		6	3	2	2	2	2	-	-	-	
s:4	HTB1	HHF2	KAR9	HHT1	MET6	HHT2	HTA2	DFG5	HTB2	TOF2	
	14	5	4	3	3	3	3	2	2	2	