HHS Public Access

Author manuscript

Proc IEEE Int Symp Bioinformatics Bioeng. Author manuscript; available in PMC 2016 November 04.

Published in final edited form as:

Proc IEEE Int Symp Bioinformatics Bioeng. 2010 ; 2010: 79–85. doi:10.1109/BIBE.2010.21.

Comparison of Co-Temporal Modeling Algorithms on Sparse Experimental Time Series Data Sets

Edward E. Allen,

Department of Mathematics, Wake Forest University, Winston-Salem, NC 27109

James L. Norris,

Department of Mathematics, Wake Forest University, Winston-Salem, NC 27109

David J. John,

Department of Computer Science, Wake Forest University, Winston-Salem, NC 27109

Stan J. Thomas, IEEE [Senior Member],

Department of Computer Science, Wake Forest University, Winston-Salem, NC 27109

William H. Turkett Jr, and

Department of Computer Science, Wake Forest University, Winston-Salem, NC 27109

Jacquelyn S. Fetrow

Department of Computer Science, Department of Physics, Wake Forest University, Winston-Salem, NC 27109

Edward E. Allen: allene@wfu.edu

Abstract

Multiple approaches for reverse-engineering biological networks from time-series data have been proposed in the computational biology literature. These approaches can be classified by their underlying mathematical algorithms, such as Bayesian or algebraic techniques, as well as by their time paradigm, which includes next-state and co-temporal modeling. The types of biological relationships, such as parent-child or siblings, discovered by these algorithms are quite varied. It is important to understand the strengths and weaknesses of the various algorithms and time paradigms on actual experimental data. We assess how well the co-temporal implementations of three algorithms, continuous Bayesian, discrete Bayesian, and computational algebraic, can 1) identify two types of entity relationships, parent and sibling, between biological entities, 2) deal with experimental sparse time course data, and 3) handle experimental noise seen in replicate data sets. These algorithms are evaluated, using the shuffle index metric, for how well the resulting models match literature models in terms of siblings and parent relationships. Results indicate that all three co-temporal algorithms perform well, at a statistically significant level, at finding sibling relationships, but perform relatively poorly in finding parent relationships.

Keywords

Biological system modeling; reverse engineering; computational algebra modeling; Bayesian modeling

I. INTRODUCTION

Analyzing experimental time series data to reconstruct networks of relationships among biological entities (genes or proteins, for example) is a challenging computational problem of immense importance [1, 2]. A number of algorithms for reverse engineering biological networks have been proposed, utilizing a variety of mathematical approaches such as relevance networks [3], graphical Gaussian models [4], Bayesian networks [5], and computational algebra [6]. For an overview of several approaches to the reverse engineering of biological networks see [7].

An important issue in modeling is the selection of a time paradigm. Broadly speaking, there are two choices for a time paradigm: next-state and co-temporal. Next-state models are commonly referenced as dynamic or Markov models. Next-state models consider data changes from one time point to the next, assuming that the underlying system can be modeled as a (typically first-order) Markov process. Next-state modeling approaches include the use of dynamic Bayesian networks [8] and state-space models [9]. Co-temporal models, on the other hand, represent mathematical relationships among the entities (genes/proteins) that exist at all time points. Modeling approaches that exploit conditional independence fall under this paradigm; this includes static Bayesian networks [5], pairwise associations (such as relevance networks [3]), or partial correlation (as in graphical Gaussian models [4]).

While both next-state and co-temporal approaches generate network graphs with vertices representing biological entities, the graphs resulting from these two time paradigms cannot be compared directly since their edges may represent different types of relationships. Next-state models result in network graphs containing directed edges representing potential causal relationships between entities. Co-temporal models often result in network graphs containing undirected edges representing correlative or co-dependency relationships.

A fundamental difficulty is that for experimental time-series data-sets which frequently only include five to ten time points, the modeling problem tends to be heavily underdetermined; thus, many network models may be consistent with the available data [10]. However, it is essential to assess algorithms on real experimental data, as simulated data does not exhibit all the features of actual data.

In this contribution, we evaluate co-temporal approaches to modeling using computational algebra and two probabilistic algorithms, continuous Bayesian and discrete Bayesian, and report the results from modeling three experimentally collected time-series datasets (two of which have replicates). Because of the nature of co-temporal algorithms, we hypothesized that these algorithms should identify particular relationships; specifically we hypothesize that co-temporal algorithms should identify sibling relationships better than parent-child relationships. The discrete Bayesian algorithms have been widely described in the literature; Pe'er [5], for example, provides a primer on discrete Bayesian modeling. The computational algebra algorithm is a variant of one originally proposed by Laubenbacher and Stigler [6, 11] that constructs polynomial equations that fit relationships among the biological entities. The continuous Bayesian modeling algorithm utilizes multivariate log-normal likelihood theory for the entities' measurement over time (see, for example, [12, 13]). The resulting

models consist of an ordered list of edges. We compare the ordered list of edges to derived literature models using the shuffle index [14] to gain insight into how each algorithm performs within the co-temporal modeling paradigm on: 1) identifying different entity relationships; 2) dealing with sparse experimental data; and 3) handling experimental noise.

II. Co-Temporal Time Paradigm for Modeling

Consider a data matrix $D=(d_{ki})$, where d_{ki} represents the measurement of entity i at time point p_k . Next-state methods produce functions $g_1, g_2, \dots, g_n$ such that $d_{k+1,i}$ or its probabilistic expectation $E(d_{k+1,i})$, equals $g_i(d_{k,1} d_{k,2} \dots d_{k,n})$. Co-temporal modeling, on the other hand, produces functions $f_1, f_2, \dots, f_n$ such that $d_{k,i}$ or its probabilistic expectation $E(d_{k,i})$ equals $f_i(d_{k,1} d_{k,2} \dots d_{k,i-1}, d_{k,i+1}, \dots, d_{k,n})$. Note that the term $d_{k,i}$ is not a variable in the function f_i . Co-temporal functions find invariants or associations in the data and predict the state of an entity at a given time point based on the values of other entities at that same time point. This function applies across all rows of the data matrix D .

The identification of co-temporal associations among the entities can be considered a generalization of clustering techniques. Discretizations of the data, as a first cut, identify similarly acting entities by giving them the same discretization. Two entities, corresponding to data columns i and j respectively, with the same discretization have the co-temporal function $x_{t,i}=x_{t,j}$. General co-temporal modeling, however, supports a broader class of functions. For example, inverse ($x_{t,i}=-x_{t,j}$) and multi-variable ($x_{t,i}=-x_{t,j}+x_{t,k}$) functions are potential co-temporal functions among biological entities that would not be identified by standard clustering techniques.

III. Modeling Algorithms

Three different modeling algorithms are evaluated – discrete Bayesian (DB), continuous Bayesian (CB), and computational algebra (CA), all implemented within a co-temporal paradigm. A brief review of each algorithm is presented below.

Both Bayesian modeling algorithms employed in this paper are based on searching the space of Bayesian networks. Bayesian networks [15, 16] are directed acyclic graphs that represent a full joint distribution through factorization into conditional probability distributions. Nodes represent variables, and the presence/absence of edges specifies dependence and independence relationships between the variables. A Bayesian network associates a conditional probability function with each variable representing the probability of the variable taking on a particular value given the value of each of its parents in the graph. In the biological context, nodes correspond to biological entities of interest and edges represent hypothesized biological dependencies between such entities (such as transcription factor regulatory control).

Our continuous Bayesian method uses the original continuous data and employs a multivariate log-normal model. Taking the logs of the levels of the original data entities transforms the original right-skewed distributions to be more symmetric and converts multiplicative chance errors to additive ones [17]. An inverse function of the Bayesian Information Criterion (BIC) is used to estimate the posterior probability of a given directed

graph [18]. This results in a directed graph's posterior probability being higher if it fits the data better (i.e., has higher maximum log likelihood), while it is lower if it is more complex (i.e., has more edges).

For any particular directed graph, its maximum log likelihood is the sum of the individual entities' maximum log likelihoods. For an entity's likelihood to be well-defined (with a non-singular sample covariance matrix for the entity and its m parents), m must be less than or equal to $T-2$, where T is the number of measured time points [19]. Accordingly, the number of edges going into or out from any node is limited to $T-2$.

Metropolis-Hastings search [20] over the space of directed graphs is employed in order to estimate probabilities of edges and graphs. This process employs $5 \times 2 = 10$ separate runs of the algorithm, where five different starting directed graphs and two different acceptance burn-in criteria are used. For each run, at least 2.5 million burn-in replications and twenty-five million regular replications are used. The list of its top two-hundred directed graphs and their proportional posterior probabilities are tabulated for each run, and an amalgamation of these ten lists is used as the basis for final posterior probability estimates for edges and graphs. The posterior probability estimate for a directed edge is the sum of the posterior probabilities for all directed graphs that contain the edge. The posterior probability estimate for an undirected edge is the sum of the posterior probability estimates for each of its two associated directed edges.

In discrete Bayesian networks, nodes in the underlying DAG represent discrete state random variables. The discrete, co-temporal Bayesian algorithm employed [21] implements Metropolis Hastings Markov Chain Monte Carlo (MCMC)-based search [20] over the space $\mathcal{M}$ of static Bayesian network structures [22]. Variations on standard MCMC structure search for discrete Bayesian networks can be found in [23–25]. MCMC over discrete Bayesian networks is implemented by a repeated process of proposing a one-edge change (addition, deletion, or reversal) to the current model, followed by acceptance or rejection of the change based on the relative likelihood and neighborhood sizes of the current and proposed models. Assuming MCMC convergence, sampling of accepted networks approaches the distribution over models $P(\text{model} / \text{data})$. Posterior probabilities of features (network model edges) are generated from counting edge occurrence over all sampled networks [5]. In this work, undirected edges are returned, with frequencies set as the sum of the corresponding directed edge frequencies.

The discrete Bayesian modeling and search parameters employed in this work are a single equal-width-bin ternary discretization, an empty initial network, a vertex fan-in limit of four, Bayesian Dirichlet equivalent (BDe) scoring, uniform priors over networks, MCMC burn-in of one million networks, and MCMC sampling of five million networks. Variables that discretize to the same vector are joined as a single combination node before graph search. In a post-processing step, these nodes are re-split into the original variables, adding edges from the resulting nodes to any nodes the combined node was connected to, and adding an additional edge, labeled with a probability of 1.0, between those variables that discretized to the same vector.

An underlying assumption of statistically driven co-temporal modeling approaches, such as Bayesian networks, is that the data collected for training represents independent samples of associations between entities.

The computational algebra algorithm uses a combination of techniques from abstract algebra and game theory, but, similar to the Bayesian approaches, is strongly based on sampling and consensus. First, several discretizations of the data are chosen. In this implementation, the choices were mean under/over [26], chi-merge 2-bin and 3-bin [27], k-means 2-bin and 3-bin [28], and k-medoids 2-bin and 3-bin [29]. For each discretization of the data, Lagrange interpolation polynomials are constructed. These polynomials are reduced using Buchberger's algorithm [30] over a sampling of orderings of the entities (genes/proteins). Finally, the Deegan-Packel index of power [31] is used to construct a power matrix. The power matrices for all of the discretizations are summed together to get a consensus power matrix M . The row k and column i entry of M gives the power score of how entity (protein/gene) i affects entity k . Finally, we set $P=M+M^T$. The matrix P is symmetric. The entries in this consensus matrix are then ranked using a Z-score. P is used to identify which entities associate most strongly with each other [32, 33].

IV. Data Sets and Literature Models

The modeling algorithms were applied to the following experimental data sets: yeast cell cycle gene expression, dendritic cell (DC) maturation gene expression, and IGF-1 signaling. The data sets represent two experimental data types - gene expression from microarrays (yeast cell cycle and DC maturation) and protein modification from Western blots (IGF-1 signaling) - and a range of time courses (minutes for the IGF-1 signaling data; hours for the yeast cell cycle data; and hours to a day for the DC maturation data). In addition, two of the experiments were performed as replicates - dye-swapped technical replicates in the case of the yeast cell cycle data and biological replicates in the case of the DC maturation data.

The yeast cell cycle data is gene expression data extracted as the signal log ratio from microarray experiments performed on yeast cells that were first synchronized by alpha factor [34, 35]. This data set consists of nine genes that are known to transcriptionally regulate progression of the cell cycle. We limit our modeling of the yeast cell cycle data sets from time $t=10$ minutes to $t=120$ minutes, as Pramila et al. report an issue in the response of the alpha synchronization of the yeast cells. The continuous Bayesian approach only models the first eight time points ($t=10$ to $t=80$). Since the computational algebra is exponential in the number of time points, the models were constructed by sampling one thousand collections of eight time points. The DC maturation data is microarray data collected following stimulation of cultured mouse bone marrow cells with poly(I:C) to represent viral stimulation [36]. It consists of 12 genes. Finally, the cell signaling data was collected from cultured chondrocytes following their stimulation with IGF-1 [37], using densitometry scans across the Western blots. This data set consisted of 11 protein sites (including isoforms/separate phosphorylation sites) that undergo phosphorylation after stimulation.

For the three experimental time-series data sets, we have extracted what is known about the networks from the literature [34, 35, 37–39], several internet sites – KEGG at

www.genome.jp/kegg/ [40] and Science STKE, at stke.sciencemag.org/cm/ [41] – and one software package – Ingenuity ® at <http://www.ingenuity.com> [42]. The literature models, shown in Fig. 1, represent the “best guess” at correct models. It is important to note that not everything is known about these models (for example, the particular details of IGF-1 signaling in chondrocytes or poly(I:C) induction of DC maturation are not understood), so the details of these models may not be perfectly correct. More details about how these models were extracted are described elsewhere [33].

V. Relationship Definitions

We assess how the co-temporal implementation of several types of algorithms can: 1) identify specific types of relationships between biological entities; 2) deal with the vagaries of sparse time course data; and 3) deal with experimental noise in replicate data sets. Of importance to accomplishing the first goal is to define the types of relationships between the entities in the models. In this work, we evaluate two types of relationships: parent-child and siblings. A parent-child relationship is one in which the parent directly affects the child entity in a time-ordered fashion. In biology, such relationships are illustrated by kinases which affect the phosphorylation of another protein, or transcriptional regulators which affect the transcription of specific genes. In Fig. 1, parent-child relationships are represented by black arrows. Siblings are those entities with a common parent. Biologically, two genes whose expression is activated by a common regulator are siblings. Likewise, two proteins phosphorylated by the same parent protein are siblings. In Fig. 1, siblings are represented by nodes of a common color. For the IGF-1 signaling network, the gray nodes represent multiple phosphorylation sites on the same protein which in this analysis are not identified as siblings since their sibling relationship is unknown. The green nodes corresponding to Shc p46, Shc 52 and Shc p66 represent isoforms of the same protein; they are not considered siblings in this analysis since they have no parents in this data set.

VI. Model Evaluation

Each of the modeling algorithms produce an ordered list of edges with associated scores. For the Bayesian approaches, these scores are edge probabilities. In the computational algebra approach, the scores are Z-scores representing strength of association from the matrix P described in Section III. While a common approach for model validation is to demonstrate links between high scoring edges (above a particular threshold) and the biological literature – examples include [3, 9, 44] – an approach using the shuffle index is used in this work to help mitigate the effects of chance matches and arbitrary threshold selection. As illustrated by the Birthday Paradox [45], matches between the top ranked edges and literature models may occur at random more often than expected; therefore, it is essential to evaluate the complete rank ordering of all edges.

The perfect output of a modeling algorithm would be an ordered list of edges in which the edges in the literature model are all ranked higher than the edges not in the literature model. When edges not in the literature model are ranked higher by an algorithm than those that are in the literature model, an inversion exists. The shuffle index scores the ordered lists produced by the modeling algorithm versus the literature model by determining the

probability of generating by a random process an ordered list with the same or fewer inversions. The shuffle index [14] is equivalent to the p-value of a one-sided Wilcoxon rank sum test [46].

It is possible for two or more edges to score the same in a given model. To score such lists, we consider such edges to be on the same row in the ordered lists. Specifically, consider the case where there are two edges E1 and E2 with the same score and E1 is in the literature model and E2 is not. Without the tie, the edge list {E1, E2} would correspond to the shuffle YN with no inversions, while the edge list {E2, E1} would correspond to the shuffle NY that has a single inversion. Under our scoring methods, the edge list that has a tie, E1/E2, has a 0.5 inversion. Such cases occur in the output of our modeling algorithms; thus, ordered lists with fractional inversions are in the Supplementary Information. Since the shuffle index is only defined for integer inversions, if the number of inversions is $X.5$, with X an integer, we return the average of the shuffle index scores for X and $X+1$ inversions, respectively.

VII. Results

The goal of this work was two-fold: to test our hypothesis that co-temporal modeling should identify sibling relationships better than parent-child relationships and to understand the strengths and weaknesses of the co-temporal implementation of various algorithms on actual experimental data. Specifically, we asked how well the algorithms 1) identify two types of entity relationships, parent and sibling, between biological entities, 2) deal with experimental sparse time course data, and 3) handle experimental noise seen in replicate data sets. All fifteen (five data sets by three algorithms) ordered edge lists generated in this work are provided in the Supplementary Information. The shuffle index scores of these lists are computed in three ways and reported in Table 1. Shuffle index scores are computed relative to the sibling relationships indicated by the literature models, relative to the parent-child relationships in the literature models, and relative to the combination of parent-child and sibling relationships in the literature models.

In interpreting the scores in the table, it is important to note that the literature models should not be considered to be the absolutely correct model of the system. The literature models incorporate information that is known about the systems generally but there may be edges and associations among the entities that are not represented in the literature models. Hence, small differences in the numbers in the table, such as the difference in the numbers given for the IGF-1 signaling in chondrocytes dataset between the discrete Bayesian (0.0101) and the computational algebra algorithms (0.0136), should not be considered significant.

First, we ask whether the co-temporal algorithms identify parent-child or sibling relationships better. Since the shuffle index scores are equivalent to the p-values of the one-sided Wilcoxon rank sum test, the scores less than or equal to 0.05 can be described as statistically significant and the scores less than or equal to 0.01 are strongly statistically significant. The results indicate that in none of the data sets modeled by these co-temporal algorithms are the parent-child relationships modeled at a level of correctness of statistical significance, while in eleven of the fifteen cases siblings are correctly returned at statistically significant levels with four of those eleven indicating strong statistical significance (Table 1).

When looking at correctness of both parents and siblings, only one case indicates a statistically significant model - computational algebra modeling of the IGF-1 phosphorylation signaling network. These results support our contention that co-temporal approaches should identify sibling relationships, and not parent-child relationships.

Next, we asked how well the three algorithms performed on sparse time course experimental data. The data in Table 1 indicate that two of the four lowest sibling scores are observed when modeling the yeast cell cycle datasets. These data sets have approximately four times the number of time points as the DC maturation and IGF-1 signaling data sets. These results support an initial intuitive hypothesis that increases in the number of available time points improve the performance of these co-temporal algorithms. The questions of whether there is an upper-bound optimal number of time points for learning and whether such an upper bound number of time points is feasible to be collected was not addressed directly by this work. However, it is also observed that only four of the fifteen models produced by the algorithms are not statistically significant, indicating that all three co-temporal algorithms appear to perform well on sparse experimental data.

Finally, we ask how well the algorithms handle experimental “noise”. To partially address this question, we compared results between replicate data sets. Three models are non-significant, with respect to identifying sibling relationships: discrete Bayesian (DB) on yeast cell cycle replicate 1; computational algebra (CA) on yeast cell cycle replicate 1; and discrete Bayesian on dendritic cell maturation replicate 2 (Table 1). These results suggest that any one data set might produce incorrect results. It is essential to perform modeling on replicate data sets. These data also suggest that discrete Bayesian methods may be more susceptible to sparse, noisy experimental data than continuous Bayesian or computational algebra algorithms. However, more modeling of experimental data sets is essential before drawing firm conclusions regarding algorithm performance on noisy experimental data.

VIII. DISCUSSION

On these five data sets, the co-temporal modeling algorithms identify siblings better than parents; none of the three algorithms identify parents well. The shuffle index scores for parent-child relationships are generally above 0.50 which indicates that creating lists of parent-child relationships by random means is as good or better than list generation by these algorithms. On the other hand, all three co-temporal algorithms generally had scores below 0.02 in identifying sibling relationships present in the literature models. Some of these scores were as low as 0.0003. Recall that if an ordered list L has a score of 0.02 then that means there is only a one in fifty chance of randomly developing a list as good or better than L . Thus, accurate interpretation of network models requires an understanding of what types of relationships a computational approach might identify. A modeling algorithm that performs well in finding parent-child relationships should generate network models that visually look similar to those in Figure 1. Algorithms that perform well in finding siblings would generate network models that primarily consisted of edges between nodes colored the same in Fig. 1. Examples of models produced by the current algorithms are shown in Fig. 2. While the networks that are shown in Fig. 2 are conservative, only highlighting the edges with the most support according to the modeling algorithms, they lend support to the

observation that co-temporal algorithms perform better at finding siblings. For the computational algebra and discrete Bayesian approaches, there are as many or more correct sibling edges in the models in Fig. 2 than there are correct parent-child edges.

There are four exceptions to the relatively good sibling scores in Table 1. Interestingly, each technique did poorly on at least one data set. The scores of the discrete Bayesian and computational algebra algorithms on the yeast cell cycle replicate 1, the scores of the discrete Bayesian algorithm on the DC maturation replicate 1 and the scores of the continuous Bayesian algorithm on the IGF-1 are all substantially higher than the other scores. In fact, since the remaining scores were all below 0.05 each of these was statistically significant.

The high shuffle index scores from the discrete Bayesian technique stood out from the scores of the other two techniques. One potential explanation is that the discrete Bayesian algorithm is a standard implementation that uses only a single discretization. The difficulty and importance of selecting discretizations appropriately has been addressed in the literature [47–49]. Different discretizations of the same data can lead to significantly different models. Hence, if the single discretization does not adequately reflect the different states of the biological entities, the resulting computational models may not be similar to the literature model. On the other hand, the computational algebra algorithm uses game theory to construct a consensus ordered list over multiple discretizations.

We modified the discrete Bayesian modeling algorithm to develop consensus models over multiple discretizations. The scores were 0.4042, 0.903, 0.7322, 0.5 and 0.2708, respectively, for identifying the sibling relationship. These results are worse than those shown in Table 1 for the Discrete Bayesian approach employing a single 3-bin discretization. Previous work by Yu et al. [50] provides evidence suggesting that discrete Bayesian network learning results in significant imprecision in returned results when only a binary discretization is used, even when there is significant number of training examples to learn from. Imprecise 2-bin results could be magnified in our consensus discretization approach, where four of the seven discretizations used are 2-bin discretizations. We plan to evaluate the results returned from learning on each different discretization to see if such a bias is evident, as understanding the sensitivity of these algorithms with respect to discretizations is an important topic.

This work highlights the importance of understanding the choice of a modeling time paradigm and how that choice affects the types of biological relationships that can be effectively identified. One cannot necessarily expect the results of a network to appear similar to those typically drawn by biologists (such as those shown in Fig. 1). Co-temporal approaches, including static Bayesian, relevance networks, and graphical Gaussian models, appear to identify sibling relationships better than parent-child relationships. Given that siblings are defined as common targets of the same parent, it is not unreasonable that they would have stronger instantaneous relationships, particularly if they are truly co-regulated by their parent(s). This notion fits well with our observed results. Because of their use of only instantaneous relationships, it can be argued that co-temporal modeling algorithms do not utilize some information that exists in time series data sets. In particular, by using only

instantaneous information, a co-temporal approach will have difficulty in finding statistical support for cause-effect relationships that are separated in time with respect to the time points observed and which do not require the cause and effect variables to have a long-term persistence in state. Arguably, there are parent-child relationships in biological systems that fit this scenario and thus can be missed by co-temporal modeling approaches. In this work, the relationships discovered by co-temporal modeling algorithms when applied to time-series data are reported. The results of next-state approaches applied to these same datasets are not presented. A judgment of the relative quality of models arising from co-temporal and from next-state approaches is not intended in this work, but rather solely a report on the properties of results obtained from co-temporal modeling algorithms. A topic for future research could be to examine relationships between co-regulated modules rather than individual entities [51]. For discrete algorithms, our observations indicate that implementation of multiple, rather than singular discretizations, is also important. Finally, these results suggest that merging models across algorithms or across replicate data sets to build on the idea of consensus modeling could be a productive approach to identifying biological networks from sparse experimental data.

Elucidating the mechanisms underlying systems-level cellular activity remains one of the most important open problems in cellular biology today. This work evaluated the applicability of co-temporal modeling to the problem of reverse engineering networks from real-world (sparse and noisy) time series datasets, with a primary intent of gaining insight into the ability of co-temporal algorithms to detect parent-child and sibling relationships. Our results suggest that the primary application of co-temporal algorithms should be in detecting sibling relationships, as statistically significant results for siblings were returned by multiple co-temporal approaches across a range of datasets. The new insights gained from this work should be useful in informing both the future choice of modeling algorithms and the future interpretation of modeling results.

Acknowledgment

The authors express their appreciation to Laura Soito for densitometric scanning of the Western blots. This work is supported by the NSF-NIGMS Program in Mathematical Biology through a grant, NIH R01-GM075304, to JSF.

References

1. Markowetz F, Spang R. Inferring cellular networks--a review. *BMC Bioinformatics*. 2007; 8(Suppl 6):S5.
2. Grzegorzczak, M.; Husmeier, D.; Werhli, A. Reverse engineering gene regulatory networks with various machine learning methods. In: Emmert-Streib, F.; Dehmer, M., editors. *Analysis of Microarray Data*. Wiley-VCH; 2008. p. 101-167.
3. Butte AJ, Tamayo P, Slonim D, Golub TR, Kohane IS. Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proc Natl Acad Sci U S A*. 2000; 97:12182–12186. [PubMed: 11027309]
4. Friedman N. Inferring cellular networks using probabilistic graphical models. *Science*. 2004; 303:799–805. [PubMed: 14764868]
5. Pe'er D. Bayesian network analysis of signaling networks: a primer. *Sci STKE*. 2005; 2005:p14.
6. Laubenbacher R, Stigler B. A computational algebra approach to the reverse engineering of gene regulatory networks. *J Theor Biol*. 2004; 229:523–537. [PubMed: 15246788]

7. Stolovitzky, G.; Califano, A. *Annals of the New York Academy of Science*. New York: Wiley-Blackwell; 2008 Jan. *Reverse Engineering Biological Networks: Opportunities and Challenges in Computational Methods for Pathway Inference*.
8. Yu J, Smith VA, Wang PP, Hartemink AJ, Jarvis ED. Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*. 2004; 20:3594–3603. [PubMed: 15284094]
9. Rangel C, Angus J, Ghahramani Z, Lioumi M, Sotharan E, Gaiba A, Wild DL, Falciani F. Modeling T-cell activation using gene expression profiling and state-space models. *Bioinformatics*. 2004; 20:1361–1372. [PubMed: 14962938]
10. Just W. Reverse engineering discrete dynamical systems from data sets with random input vectors. *J Comput Biol*. 2006; 13:1435–1456. [PubMed: 17061920]
11. Stigler, B. *Mathematics*. Blacksburg: Virginia Polytechnic Institute and State University; 2005. *An Algebraic Approach to Reverse Engineering with an Application to Biochemical Networks*; p. 76
12. John, D.; Fetrow, JS.; Norris, J. Metropolis-Hastings algorithm and continuous regression for finding next-state models of protein modification using information scores. In: Yang, J.; Yang, M.; Zhu, M.; Zhang, Y.; Arabnia, H.; Deng, Y.; Bourbakis, N., editors. *Proceedings of the 7th International Symposium on Bioinformatics and Bioengineering*. Vol. I. Boston, MA: IEEE; 2007 Oct. p. 35-41.
13. John D, Fetrow JS, Norris J. Continuous Co-temporal Probabilistic Modeling of Systems Biology Networks from Sparse Data. Unpublished.
14. Allen, E.; Diao, L.; Fetrow, JS.; John, DJ.; Loeser, RF.; Poole, LB. The shuffle index and evaluation of models of signal transduction pathways. In: Burg, J., editor. *Proceedings of the 45th Annual Association of Computing Machinery Southeast Conference*. Winston-Salem, NC: ACM; 2007 Mar. p. 250-255.
15. Charniak E. Bayesian networks without tears: making Bayesian networks more accessible to the probabilistically unsophisticated. *AI Magazine*. 1991; 12:50–63.
16. Jensen, F. *Bayesian Networks and Decision Graphs*. New York: Springer-Verlag; 2001.
17. Neter, J.; Wasserman, W.; Kutner, M. *Applied Linear Statistical Models*. 2nd ed. Irwin Professional Publishing; 1985.
18. Schwarz G. Estimating the dimension of a model. *Annals of Statistics*. 1978; 6:461–464.
19. Johnson, R.; Wichern, D. *Applied Multivariate Statistical Analysis*. Prentice-Hall; 1982.
20. Hastings W. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*. 1970; 57:97–109.
21. Allen, E.; Pecorella, A.; Fetrow, JS.; John, D.; Turkett, W. Reconstructing networks using co-temporal functions. In: Silaghi, M., editor. *Proceedings of the 44th Annual Association of Computing Machinery Southeast Conference*. Melbourne, FL: ACM; 2006 Mar. p. 417-422.
22. Madigan D, York J. Bayesian graphical models for discrete data. *International Statistical Review*. 1995; 63:215–232.
23. Husmeier D. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*. 2003; 19:2271–2282. [PubMed: 14630656]
24. Friedman N, Koller D. Being Bayesian about network structure. *Machine Learning*. 2003; 50:95–126.
25. Grzegorzczak M, Husmeier D. Improving the structure MCMC sampler for Bayesian networks by introducing a new edge reversal move. *Machine Learning*. 2008; 71:265–305.
26. Catlett J. On changing continuous attributes into ordered discrete attributes. *European Working Session on Learning*. 1991
27. Kerber, R. Chi-merge: discretization of numeric attributes; presented at Ninth International Conference on Artificial Intelligence; 1992.
28. Everitt, B. *Cluster Analysis*. New York: Halsted Press; 1974.
29. Kaufman, L.; Rousseeuw, P. Clustering by means of medoids. In: Dodge, Y., editor. *Statistical Data Analysis Based on the L1-Norm and Related Methods*. North-Holland; 1987. p. 405-416.
30. Cox, D.; Little, J.; O'Shea, D. *Ideals, Varieties, and Algorithms*. New York: Springer-Verlag; 1992.

31. Deegan J, Packel E. A new index for simple n-person games. *International Journal of Game Theory*. 1978; 7:113–123.
32. Allen EE, Fetrow JS, Daniel LW, Thomas SJ, John DJ. Algebraic dependency models of protein signal transduction networks from time-series data. *J Theor Biol*. 2006; 238:317–330. [PubMed: 16002094]
33. Allen E, John D, Turkett W, Norris J, Thomas S, Hiltbold E, Daniels L, Loeser RF, Nelson K, Poole LB, Fetrow JS. Next-state and co-temporal modeling of time-series data using computational algebra techniques. Unpublished.
34. Pramila T, Miles S, GuhaThakurta D, Jemiole D, Breeden LL. Conserved homeodomain proteins interact with MADS box protein Mcm1 to restrict ECB-dependent transcription to the M/G1 phase of the cell cycle. *Genes Dev*. 2002; 16:3034–3045. [PubMed: 12464633]
35. Pramila T, Wu W, Miles S, Noble WS, Breeden LL. The Forkhead transcription factor Hcm1 regulates chromosome segregation genes and fills the S-phase gap in the transcriptional circuitry of the cell cycle. *Genes Dev*. 2006; 20:2266–2278. [PubMed: 16912276]
36. Inaba K, Inaba M, Romani N, Aya H, Deguchi M, Ikehara S, Muramatsu S, Steinman RM. Generation of large numbers of dendritic cells from mouse bone marrow cultures supplemented with granulocyte/macrophage colony-stimulating factor. *J Exp Med*. 1992; 176:1693–1702. [PubMed: 1460426]
37. Starkman BG, Cravero JD, Delcarlo M, Loeser RF. IGF-I stimulation of proteoglycan synthesis by chondrocytes requires activation of the PI 3-kinase pathway but not ERK MAPK. *Biochem J*. 2005; 389:723–729. [PubMed: 15801908]
38. Takaoka A, Yanai H. Interferon signalling network in innate defence. *Cell Microbiol*. 2006; 8:907–922. [PubMed: 16681834]
39. Bonjardim CA. Interferons (IFNs) are key cytokines in both innate and adaptive antiviral immune responses--and viruses counteract IFN action. *Microbes Infect*. 2005; 7:569–578. [PubMed: 15792636]
40. Kanehisa M, Araki M, Goto S, Hattori M, Hirakawa M, Itoh M, Katayama T, Kawashima S, Okuda S, Tokimatsu T, Yamanishi Y. KEGG for linking genomes to life and the environment. *Nucleic Acids Res*. 2008; 36:D480–D484. [PubMed: 18077471]
41. Gough NR. Science's signal transduction knowledge environment: the connections maps database. *Ann N Y Acad Sci*. 2002; 971:585–587. [PubMed: 12438188]
42. I. Systems. Ingenuity Pathways Analysis. 2008 Dec. <http://www.ingenuity.com>.
43. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*. 2003; 13:2498–2504. [PubMed: 14597658]
44. Luna IT, Huang Y, Yin Y, Padillo DP, Perez MC. Uncovering gene regulatory networks from time-series microarray data with variational Bayesian structural expectation maximization. *EURASIP J Bioinform Syst Biol*. 2007:71312. [PubMed: 18309364]
45. Stallings, W. *Cryptography and Network Security*. 4th ed. Prentice-Hall; 2006.
46. Hollander, M.; Wolfe, D. *Nonparametric Statistical Methods*. New York: Wiley-Interscience; 1973.
47. Pensa, R.; Leschi, C.; Besson, J.; Boulicaut, J. Assessment of discretization techniques for relevant pattern discovery from gene expression data; presented at Fourth Workshop on Data Mining in Bioinformatics (BIOKDD04); 2004.
48. Madeira, S.; Oliveira, A. An evaluation of discretization methods for non-supervised analysis of time-series gene expression data. Instituto de Engenharia de Sistemas e Computadores Investigacao e Desenvolvimento, Technical Report 42. 2005 Dec.
49. Steck, H.; Jaakkola, T. Predictive discretization during model selection. *Pattern Recognition; 26th DAGM Symposium Proceedings, Lecture Notes in Computer Science*; Tubingen, Germany: Springer Berlin/Heidelberg. 2004. p. 1-8.
50. Yu J, Smith V, Wang P, Hartemink A, Jarvis E. Advances to Bayesian Network Inference for Generating Causal Networks from Observational Biological Data. *Bioinformatics*. 2004; 20:3594–3603. [PubMed: 15284094]

51. Segal E, Shapira M, Regev A, Pe'er D, Botstein D, Koller D, Friedman N. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*. 2003; 34(no. 2):166–276. [PubMed: 12740579]

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Fig. 1.

Literature models of the three data sets utilized in this work (in clock-wise order, from top left): yeast cell cycle, DC maturation, and IGF-1 signaling. Black arrows represent parent child relationships; gray arrows represent ancestor relationships; common colored nodes represent sibling relationships. Figures were created using Cytoscape [43].

Fig. 2.

Models produced by the three co-temporal algorithms on the yeast second replicate data set. Models in clock-wise order from the top left are from the computational algebra, continuous Bayesian, and discrete Bayesian algorithms. For the algebraic approach, edges with Z-scores above 1.5 are darkened. For the Bayesian approaches, edges with a probability above 90% are darkened. For the edges above these thresholds, the thickness and darkness of the line indicate higher scores.

Table 1

Shuffle index scores, representing the probability of getting the respective literature ordered list through random generation, for the different co-temporal methods on the five datasets of interest indicate that the best scores occur for correctly finding siblings and show a relationship with the number of available observations.

Method	Data Set	Sibling	Parents	Sibling + Parents
CB	Y Rep 1	0.0126	0.9983	0.8684
DB	Y Rep 1	0.3204	0.7691	0.6424
CA	Y Rep 1	0.1647	0.9918	0.9116
CB	Y Rep 2	0.0107	0.7738	0.1036
DB	Y Rep 2	0.0008	0.8869	0.0775
CA	Y Rep 2	0.0003	0.9883	0.3132
CB	DC Rep 1	0.0031	0.9971	*
DB	DC Rep 1	0.3368	0.6679	*
CA	DC Rep 1	0.0141	0.9863	*
CB	DC Rep 2	0.0090	0.9913	*
DB	DC Rep 2	0.0311	0.9698	*
CA	DC Rep 2	0.0167	0.9839	*
CB	IGF-1	0.1831	0.2993	0.1284
DB	IGF-1	0.0101	0.9711	0.6998
CA	IGF-1	0.0136	0.1598	0.0075