Improved Search for Structure Learning of Large Bayesian Networks

Technical Report 07-38

Avi Herscovici Oliver Brock
Department of Computer Science
University of Massachusetts Amherst

July 9, 2007

Abstract

The problem of Bayesian network structure learning is defined as an optimization problem over the space of all possible network structures. For low-dimensional data, optimal structure learning approaches exist. For high-dimensional data, structure learning remains a significant challenge. Most commonly, approaches to high-dimensional structure learning employ a reduced search space and apply hill climbing methods to find high-scoring network structures. But even the reduced search space contains many local optima so that local search methods are unable to find near-optimal network structures. Instead of focusing on search space reduction, as most of the previous work in this area, we propose to replace the greedy search schemes with more effective search methods. We show that for high-dimensional data the proposed search method finds significantly better structures than other leading approaches to structure learning.

1 Introduction

High-dimensional data sets are abundant in a variety of scientific domains, including genetics, social networks, and astronomy. To gain insight into the processes that generate this data, it is of interest to extract statistical dependencies between variables in the data. Based on these dependencies, it is possible to postulate a causal relationship. For example, we can formulate hypotheses about gene regulation by understanding statistical dependencies in gene expression data [10]. An effective method for extracting such causal relationships is to learn the structure of a Bayesian network [22] from the data.

Structure learning for Bayesian networks is commonly cast as an optimization problem. Within the space of all possible network structures, we search for the structure that explains the data best. A scoring function [17, 14] is used to evaluate the quality of the network, given the data.

For data sets of low dimensionality, the structure learning problem can be solved optimally [16, 15, 24]. In higher dimensions, however, it becomes increasingly difficult, as the number of possible network structures grows super-exponentially in the dimensionality of the data [23]. Structure learning for high-dimensional data sets thus remains a substantial challenge with far-reaching implications for practical applications.

To make structure learning tractable in high-dimensional spaces, the majority of research has focused on reducing the size of the search space. Within this reduced search space, existing methods employ local search, such as hill climbing or Monte Carlo methods, to determine high-quality structures. Some approaches augment local search with tabu lists or with random restarts to escape the numerous local optima in the scoring function. But local search remains unable to adequately search the reduced search spaces. As a consequence, many existing structure learning approaches learn network structures that only partially capture the true dependencies present in the data.

We propose to advance Bayesian network structure learning by employing an improved search method. This search method, which we call model-based search (MBS) [4], combines the computational efficiency of local search with the advantageous properties of global search. If we view local search as exploitation and global search as exploration, model-based search effectively varies the trade-off between exploration and exploitation as search progresses. Initially, when no information about the quality function is available, model-based search resorts to pure exploration. However, as this exploration yields information about the quality function, model-based search identifies areas of the search space that are likely to contain high-quality network structures and focuses search on these areas. In doing so, MBS adaptively trades off exploration and exploitation. This adaptive trade-off significantly improves the effectiveness of search in high-dimensional spaces.

Our experimental results show that structure learning using model-based search outperforms other leading structure learning approaches. On the Gene data set [25] with 801-dimensional data, for example, our method produces on average a network that is 50.7% more accurate than the structure obtained with another leading structure learning method (compared to the network that was used to generate the data). This represents a significant improvement in the state of the art of Bayesian network structure learning. We believe that the development of even better search methods is a most promising future direction for Bayesian network structure learning.

2 Related Work

We propose to advance Bayesian network structure learning by employing an improved search method. In this section we review related work in structure learning and in search.

2.1 Structure Learning

The literature in structure learning is extensive. Here, we focus on work that reduces the search space to facilitate optimization by local search. We also discuss methods that have been applied to high-dimensional data sets.

The equivalence class search space [6] represents structures as partially directed graphs (PDAGs). This search space results in relatively smooth quality landscape and can thus be searched well with greedy methods. Nevertheless, the PDAG search space still contains many local optima. Some of them can be overcome by including randomness into the search [20].

An alternative search space considers orderings of nodes, which in turn limit the connectivity of nodes in the network. The optimization problem now consists of determining the best ordering of nodes. A particular ordering is scored based on the most likely structure that observes the ordering, given the data. In prior work, a number of different search strategies were employed to find good orderings; these strategies include genetic algorithms [18], an optimized hill climbing method [26], and Markov chain Monte Carlo [9].

The search space can also be reduced by limiting the connectivity of the graph. The sparse candidate algorithm [11] only considers connections between variables with high mutual information. This algorithm was successfully applied to the Gene data set with 801 variables [25]. This idea was later extended to include a sound method for learning parent and children sets [28]. The resulting method, Max-Min Hill Climbing (MMHC), compares favorably with the sparse candidate algorithm when applied to large data sets [28]. Both of these methods rely on local search (hill climbing with a tabu list). As a consequence, they remain susceptible to local optima.

Another structure learning algorithm focuses on situations in which only very sparse data is available. The algorithm relies on frequent sets to learn large networks [13].

Although unrelated to the work presented here, it is interesting to mention another application of active learning to structure learning. Tong and Koller use active learning to obtain accurate structures from as little training data as possible [27].

2.2 Search

The literature on search and optimization is vast and we cannot hope to adequately review it here. Instead, we focus on those search methods that have been applied in structure learning and on optimization methods that are similar to the one we will propose in Section 3.

Most structure learning approaches rely on local search methods, such as hill climbing and Monte Carlo methods. To alleviate the problems that arise from local optima, these methods have been augmented with tabu lists [12] or random re-starts. These measures somewhat reduce the susceptibility to local search but fail to be effective in very large search spaces.

In very large search spaces local search is inadequate and global search is intractable. To overcome this dilemma, it is possible to bias search (or optimization) towards promising areas of the search space. If no a priori information is available, this can be achieved with an active learning-like approach: as search progresses, the obtained information is used to direct further search. The use of this information to guide search has to rely on assumptions about the underlying search space. In search spaces that do not satisfy these assumptions, these algorithms will perform poorly [29]. In practice, however, several algorithms have shown that this active learning approach to search can achieve good results.

The STAGE algorithm [3] uses the information obtained during local search to guide future local searches. STAGE learns a function that predicts the outcome of local searches started in a particular region of the search space. By biasing the starting point selection towards promising regions, STAGE can outperform other global optimization algorithms in a variety of problem domains.

The MIMIC algorithm [8] uses information obtained during global sampling to determine a probability distribution over the search space. This distribution estimates the probability that the optimization criterion evaluates to at least a certain value for a particular point in the search space. By sampling the search space based on this distribution and only retaining samples of increasing quality, the estimate of the distribution is successively refined until the algorithm converges.

An adaptive importance sampling algorithm [21] actively learns the optimal importance sampling distribution from the previously taken samples. This algorithm has been applied to action evaluation in influence diagrams.

These three aforementioned algorithms share the basic insight with our algorithm, which will be introduced in the next section: they actively learn which regions of the search space are promising. However, they fundamentally differ from our algorithm in their ability to scale to high-dimensional problems. The algorithms discussed in this section attempt to learn a function over the entire search space. This is not only computationally expensive but also requires significant amounts of training data. In contrast, our algorithm quickly reduces the search space that is considered and subsequently only attempts to estimate the properties of selected regions. As we will see in Section 4, this will allow us to perform structure learning for very high-dimensional data sets.

3 Improved Search for Structure Learning

Our goal is to devise a search technique that adequately searches very high-dimensional search spaces. Ideally, it would maintain the computational efficiency of local search methods while achieving some of the search characteristics of global methods. Since an exhaustive search of search space is impossible, we propose to devise a search method that very quickly rules out large amounts of search space and focuses search on the remaining regions. Of course, there is a significant risk to rule out the region of the search space that contains the optimum. Since the proposed search method is of heuristic nature, experimental validation will have to demonstrate that this does not happen.

In this paper, we adapt model-based search (MBS) [4] to the problem of structure learning. MBS was originally developed in the domain of protein structure prediction, which can also be cast as a high-dimensional optimization problem.

MBS is based on the assumption that a coarse sampling of a region of the search space reveals coarse information about the underlying structure of that region. This structural information is used to build an approximate model of interesting regions of the search space. MBS iteratively refines its model at it continues to explore the regions of the search space captured by the model. Effectively, search is focused on these regions and all other regions are discarded. Since search in MBS proceeds stochastically, however, discarded regions can be re-discovered.

The key to the efficiency of MBS is the simplicity of its model. The model consists of a small set of high-quality samples, a distance metric imposed on the space, and information that can be computed very efficiently using the previous two parts of the model. We believe the simplicity of the MBS's model and the focus on small regions of the search space is compelling, as more complex models for high-dimensional spaces require significant amounts of training data to be accurate. And it is also this simplicity of the model that separates MBS from the work discussed in Section 2.2. The approaches discussed in that section attempt to learn a model for the entire search space or to estimate probability distributions over that space. We believe this to be impractical for high-dimensional spaces.

Another interesting property of MBS is its combination of a global representation (the model) and local search (used in sampling). The model captures explorative aspects of search, whereas local search is exploitative. As MBS proceeds with the search, the trade-off between exploration and exploitation is continuously adjusted based on the information obtained about the search space. This adaptive trade-off results in effective search, in particular in high-dimensional spaces.

The adaptation of MBS to structure learning proceeds as follows (details are given below):

1. Create the initial model from n random samples
2. Rank the samples with the Sample Score heuristic
3. Reallocate the k worst samples to the regions represented by each member of the best $n - k$ samples
4. Sample in these regions using only the allocated amount of samples; this maintains the model size
5. Check for convergence; if optimal network does not improve, terminate; otherwise go to step 2

In the case of Bayesian networks, each sample is a DAG. In the remainder of the section we provide implementation details regarding model-based search applied to Bayesian network structure learning.

Distance Metric: We define the distance metric between two networks G and G' as the amount of local operations (edge additions, deletions, or reversals) needed to transform G into G'. The computation of the distance metric is done efficiently by accumulating the difference between the adjacency matrices of G and G' and discounting reversed edges. The distance metric imposes structure on the search space. This structure is crucial for MBS. The distance metric should create a quality landscape in which small distances between samples frequently correspond to relatively small changes in overall quality of the samples.

Sample Quality: MBS evaluates the quality of a samples s as follows:

$$\mathrm{Q}(s) = \alpha \ \mathrm{BDeu}(s) + \beta \ \mathrm{Radius}(s)$$

The overall score of a sample s depends on its BDeu score and the estimated radius of the high-quality region around s. The radius is estimated by the closest lower-scoring sample. We refer to the tuple $(s, \mathrm{Radius}(s))$ as a region of the search space. The set of all regions makes up the model of MBS. Note that by varying the ratio of α/β we can affect the exploration/exploitation trade-off made by MBS. Values larger than 1 increase exploitative behavior, whereas values smaller than one focus search on large (and thus relatively unexplored) regions.

Allocating Samples: MBS assigns samples to a particular region with the constraint that the model size must remain constant. Let $S = \{s_1, s_2, ..., s_{n-k}\}$ be the set of samples s_i that represent the $n - k$ best regions in the search space. The amount of samples allocated to the region associated with sample s_i is computed as follows:

$$\mathrm{Samples}(s_i) = \ n \, \frac{Q(s_i)}{\sum\limits_{j=1}^{n-k} Q(s_j)}.$$

Each region is assigned a fraction of the n samples based on its contribution to the total score of all the regions. Note that this score includes both the BDeu score of s as well as the estimated size of its region.

Sampling: In order to sample in the allocated regions, we start a Metropolis Hastings Monte Carlo [19] (MHMC) run from the sample representing the region. MHMC performs a random walk which will accept moves that reduce the score with a probability based on the Boltzmann equation. The Monte Carlo run is followed by hill climbing to locally optimize the sample and thus improve the quality of the information obtained from it.

4 Experiments

In this section we present experiments that validate the proposed search algorithm in two ways. First, we demonstrate that the simple model determined by MBS focuses on good regions of the search space. The resulting search outperforms repeated local search runs with an equal amount of computational resources. Second, we perform a variety of structure learning experiments and compare the performance of MBS with state-of-the-art structure learning algorithms.

4.1 Accuracy of Model

To show that the simple model maintained by model-based search enables search to focus on promising regions of the search space, we compare MBS with local search. Both algorithms are given the same amount of computational resources. An advantage in performance thus has to derive from exploiting the model to focus search on good regions of search space.

The local search method used in this experiment consists of the same blend of greedy and stochastic search that MBS uses to sample locally. By using 10 samples for MBS and comparing it with 10 independent runs of the local search, we ensure that both methods use the same amount of computation. To make this comparison fair, we generate ten random initial samples and use them for the initial model of 10-sample MBS and as the initial samples for 10 runs of local search. We use a data set generated from the Gene network [11] (801 variables).

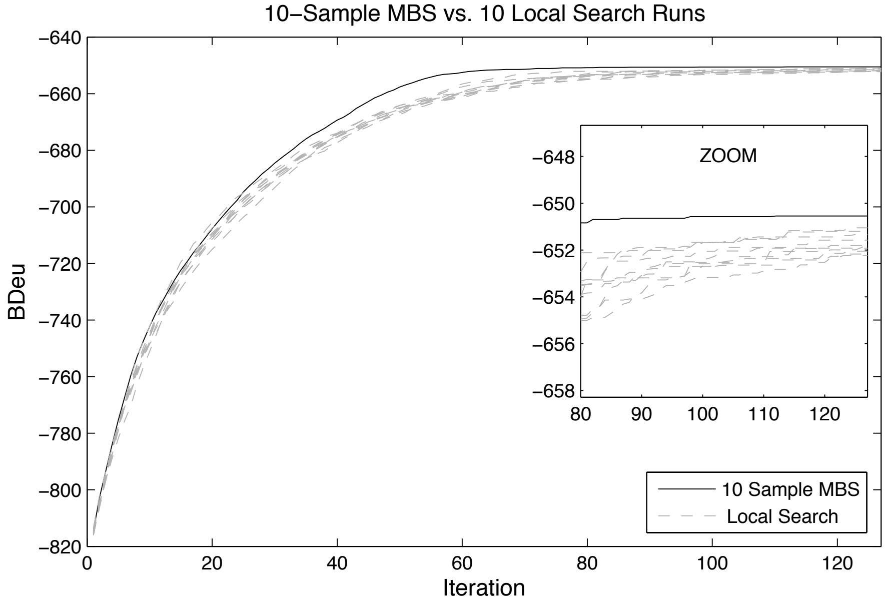

Figure 1: The graph compares the score of a 10-sample MBS run with the best score of 10 independent, local searches through 120 iterations. The quality of the structures obtained with MBS quickly dominate those obtained by local search. One of the local searches temporarily finds a better network. As search progresses, MBS outperforms each of the 10 individual searches. This demonstrates that MBS successfully focuses on good regions of the search space and outperforms local search with random restarts when given the same computational resources.

The graph in Figure 1 compares the performance of 10-sample MBS to 10 independent runs of local search. MBS learns higher-quality network structures almost throughout the entire experiment. At a point early during the search, one of the 10 independent runs temporarily outperforms MBS. This can be explained with the exploitative nature of local search. In contrast, MBS combines exploration with exploitation and thus spends computational resources in multiple regions of the search space. After 30 iterations, the explorative efforts of MBS pays off. As all of the local searches converge towards local optima, MBS converges towards a higher-quality local optimum.

This experiment demonstrates that MBS is able to successfully trade off exploration and exploitation during the search in a high-dimensional search space. The model used by MBS successfully discriminates between poor and promising regions of the search space and thus can render search more effective.

4.2 Structure Learning

We now compare the performance of MBS to leading Bayesian network structure learning algorithms. For this comparison we used six data sets. The first four (data 1 through data 4 in Figure 2) were generated using the Child10 network (10 connected copies of the Child network [7], as described in [28]). The Child10 network contains 220 nodes connected by 257 edges. The fifth data set was generated from the Alarm10 network (10 connected copies of the Alarm network [2]) with 370 variables and 570 edges. The sixth data set was generated from the Gene network (as constructed in [11]), which contains 801 variables and 972 edges. Every data set is made up of 1000 instances generated by the respective network.

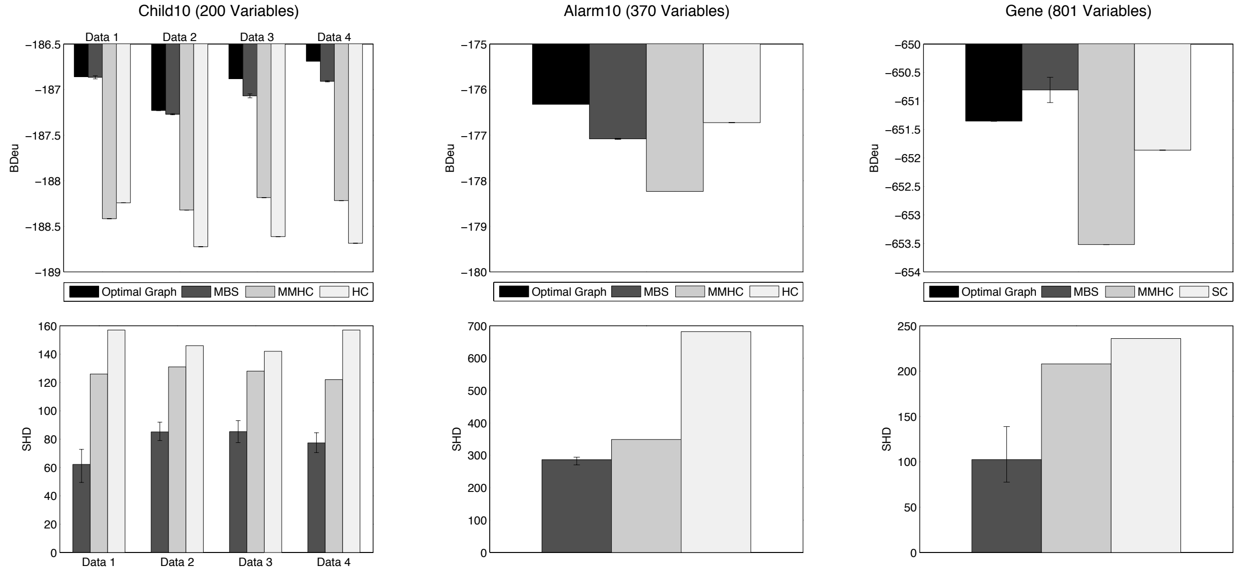

Figure 2: The performance of MBS compared to Max-Min Hill Climbing (MMHC), Hill Climbing with tabu list (HC), and Sparse Candidate (SC on Gene). The top row shows the BDeu score achieved by each algorithm, and the second row shows the SHD between the learned structure and the network used to generate the data. The mean of 100 MBS runs is shown along with one standard deviation as an error bar. MBS learns structures that have up to 50% fewer structural errors than the structure learned by the next best algorithm.

We compare the performance of MBS with three structure learning algorithms. For the first five data sets, we compare to hill climbing with a tabu list (optimized hill climbing [28], HC in Figure 2) and to optimized hill climbing in the max-min parents and children search space (MMHC). For the Gene data set, the largest in our experiment, the HC algorithm does not converge within a reasonable amount of time. For this data set we compare MBS to MMHC and the sparse candidate algorithm [11] with parameter k=5 (SC). A recent study of structure learning algorithms identified MMHC and SC as the leading structure learning algorithms for high-dimensional data [28]. The implementations of algorithms other than MBS were obtained from Causal Explorer [1]. Note that HC operates in the full search space (and thus cannot be applied to the high-dimensional Gene data), whereas MBS and MMHC operate on the reduced max-min parents and children (MMPC) search space [28]. HC operates in a third kind of search space that limits the predecessors considered for every node in the graph [11].

Since MBS is a stochastic algorithm, we average all MBS results over 100 runs with a single standard deviation marked by the error bars in Figure 2. In each experiment we use the following settings for MBS: each iteration of MBS attempts 60 Monte Carlo steps, followed by 10 Hill Climbing steps; the model of MBS contains 10 samples; $\alpha = 1$; $\beta = 0.5$. The settings for MBS were optimized for a data set different from the ones used for experimental validation in this section.

To compare the performance of the structure learning algorithms, we use two different scores for the network. In addition to the standard BDeu score [14], we also evaluate the learned structure based on its structural Hamming distance (SHD) [28] to the network used to generate the data. The structural Hamming distance is a similarity metric between two PDAGs that measures how many local moves are necessary to convert one PDAG into the other. We convert the DAGs returned by the structure learning algorithms into their equivalent PDAGs [5]. The comparison of PDAGs is useful, since Bayesian networks can be statistically equivalent yet structurally different, formally known as Markov equivalent.

We now compare the structure learning performance of the different algorithms on the six data sets. Figure 2 shows in the top row the BDeu score and in the bottom row the structural Hamming distance of the learned networks for each of the data sets. In the graphs with BDeu score, we also show the score of the network that generated the data (optimal graph).

For the Child10 data set, MBS outperforms both MMHC and HC significantly for all data sets, both in terms of BDeu score as well as in terms of SHD. On average, MBS learns a network with **50.7%** less structural errors than MMHC on the first data set and 35%, 33.4% and 36.6% less error on the other three data sets. The variance for MBS runs is very small, indicating that MBS searches the search space consistently well.

For the Alarm10 data set, MBS outperforms both MMHC and HC in terms of SHD. HC obtains a better BDeu score than MBS, but performs significantly worse in terms of SHD. This can be explained by the fact that HC searches an unconstrained search space rather than the MMPC search space. It is thus able add edges that improve the score but are not contained in the MMPC search space. The better SHD score of MBS shows that these edges do not correspond to edges in the original network. On average, MBS learns a network with 18% fewer structural errors when compared to MMHC.

For the Gene data set MBS overfits the data, as the BDeu score obtained by MBS exceeds that of the optimal network. More importantly, however, MBS outperforms both MMHC and SC significantly. On average, MBS finds a network with **50.7%** less structural errors than the next best algorithm. When we consider the best network (SHD of 34) found by MBS over 100 trials, it contains **83.7%** less structural errors that the one obtained with MMHC (SHD of 208). The worst network found by MBS (SHD of 182) still contains 12.5% less structural errors than MMHC. Relative to Alarm10, the variance of MBS is increased, indicating that MBS is not able to search the entire space as consistently as in the Alarm10 experiments. We view this as an indication that there is room for further improvement in search.

These experimental results clearly demonstrate that structure learning based on MBS significantly outperforms existing structure learning methods on the standard benchmarks for structure learning. It shows that the model used in MBS can successfully focus on good regions in the search space by adjusting the trade-off between exploration and exploitation.

5 Conclusion

We show that Bayesian network structure learning for high-dimensional data can be improved significantly by employing better search methods. Traditionally, structure learning is performed with local search methods, such as hill climbing or Monte Carlo methods. In contrast, we proposed and validated the use of a search method that combines local and global aspects of search in an active learning-based framework. The proposed search method, model-based search (MBS), uses information obtained from a sparse sampling of the search space to identify regions that are likely to contain high-quality networks. It then successively refines its understanding of the search space, focusing on increasingly narrow portions of the space. By quickly ruling out large amounts of the search space, search becomes efficient and effective, even in high-dimensional spaces.

We validated the proposed approach to Bayesian network structure learning in a variety of experiments. These experiments include the Gene data set, which includes 801 variables and has served as a benchmark problem for structure learning in high-dimensional spaces. Our method consistently outperforms existing approaches to structure learning by a large margin. In the Gene data set, our approach reduces the structural difference between the learned network and the network used to generate the training data by over 50% on average. We view these results as evidence that significant performance improvements can be obtained by further refining the search methods employed for structure learning.

Acknowledgments

This work is supported in part by the National Science Foundation (NSF) under grant IIS 0545934 and by the National Institutes of Health (NIH) under grant NIGMS 1R01GM076706. We are grateful for this support. We would like to thank Laura Brown of Vanderbilt's Discovery Systems Lab for her help with Causal Explorer and the Max-Min Hill Climbing algorithm.

References

[1] Constantin F. Aliferis, Ioannis Tsamardinos, Alexander R. Statnikov, and Laura E. Brown. Causal explorer: A causal probabilistic network learning toolkit for biomedical discovery. In *Intl. Conf. on Mathematics and Engineering Techniques in Medicine and Biological Sciences*, pages 371–376, 2003.

[2] I.A. Beinlich, H.J. Suermondt, R.M. Chavez, and G.F. Cooper. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *Proc. of the Second European Conf. on Artificial Intelligence in Medicine*, pages 247–256, 1989.

[3] Justin Boyan and Andrew Moore. Learning evaluation functions for global optimization and boolean satisfiability. In *Fifteenth National Conf. on Artificial Intelligence*, 1998.

[4] TJ Brunette and Oliver Brock. Improving protein structure prediction with model-based search. *Bioinformatics*, 21:66–74, 2005.

[5] David Chickering. A transformational characterization of equivalent Bayesian network structures. In *Proc. of the 11th Annual Conf. on Uncertainty in Artificial Intelligence (UAI-95)*, pages 87–98, San Francisco, CA, 1995.

[6] David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2003.

[7] Robert G. Cowell, Steffen L. Lauritzen, A. Philip David, David J. Spiegelhalter, and David J. Spiegelhater. *Probabilistic Networks and Expert Systems*. Springer-Verlag, 1999.

[8] J. De Bonet, C. Isbell, and P. Viola. Mimic: Finding optima by estimating probability densities. In Michael Jordan, Michael Mozer, and Michael Perrone, editors, *Advances in Neural Information Processing*, volume 9, Denver 1996, 1996. MIT Press, Cambridge.

[9] Nir Friedman and Daphne Koller. Being Bayesian about network structure. A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning*, 50(1-2), January 2003.

[10] Nir Friedman, Michal Linial, Iftach Nachman, and Dana Peér. Using Bayesian networks to analyze expression data. In *RECOMB '00: Proc. of the Fourth Annual Intl. Conf. on Computational Molecular Biology*, pages 127–135, 2000.

[11] Nir Friedman, Iftach Nachman, and Dana Peér. Learning Bayesian network structure from massive datasets: The "sparse candidate" algorithm. In *Proc. of the 15th Annual Conf. on Uncertainty in Artificial Intelligence*, pages 206–215, San Francisco, CA, 1999.

[12] Fred Glover and Fred Laguna. *Tabu Search.* Kluwer Academic Publishers, 1997.

[13] Anna Goldenberg and Andrew Moore. Tractable learning of large Bayes net structures from sparse data. In *Proc. of the Twenty-First Intl. Conf. on Machine Learning*, page 44, 2004.

[14] David Heckerman, Dan Geiger, and David M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.

[15] Mikko Koivisto. Advances in exact Bayesian structure discovery in Bayesian networks. In *Proc. of the 22nd Annual Conf. on Uncertainty in Artificial Intelligence*, San Francisco, CA, 2006.

[16] Mikko Koivisto and Kismat Sood. Exact Bayesian structure discovery in Bayesian networks. *Journal of Machine Learning Research*, 5:549–573, 2004.

[17] W. Lam and F. Bacchus. Learning Bayesian belief networks: An approach based on the mdl principle. *Computational Intelligence*, 10(3):269–293, 1994.

[18] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murgam, and C. M. H. Kuijpers. Structure learning of Bayesian networks by genetic algorithms: A performance analysis of control parameters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(9):912–926, 1996.

[19] N. Metropolis and S. Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44:335–341, 1949.

[20] Jens Nielsen, Tomas Kocka, and Jose Peña. On local optima in learning Bayesian networks. In *Proc. of the 19th Annual Conf. on Uncertainty in Artificial Intelligence*, pages 435–442, San Francisco, CA, 2003.

[21] Luis Ortiz and Leslie Kaelbling. Adaptive importance sampling for estimation in structured domains. In *Proc. of the 16th Annual Conf. on Uncertainty in Artificial Intelligence*, pages 446–454, San Francisco, CA, 2000.

[22] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufman Publishers, 1988.

[23] R. W. Robinson. Counting labeled acyclic digraph. In F. Harary, editor, *New Directions in Graph Theory*, pages 239–273, 1973.

[24] Tomi Silander and Petri Myllymki. A simple approach for finding the globally optimal Bayesian network structure. In *Proc. of the 22nd Annual Conf. on Uncertainty in Artificial Intelligence*, San Francisco, CA, 2006.

[25] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol Biol Cell*, 9(12):3273–3297, December 1998.

[26] M. Teyssier and D. Koller. Ordering-based search: A simple and effective algorithm for learning Bayesian networks. In *Proc. of the Twenty-first Conf. on Uncertainty in AI*, pages 584–590, Edinburgh, Scottland, UK, July 2005.

[27] Simon Tong and Daphne Koller. Active learning for structure in Bayesian networks. In *Proc. of the Intl. Joint Conf. on Artificial Intelligence*, Seattle, USA, 2001.

[28] I. Tsamardinos, L.E. Brown, and C.F. Aliferis. The Max-Min Hill-Climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–79, 2006.

[29] David H. Wolpert and William G. Macready. No free lunch theorems for search. Technical Report SFI-TR-95-02-010, Santa Fe Institute, Santa Fe, USA, 1995.