

Feature-inclusion Stochastic Search for Gaussian Graphical Models

BY JAMES G. SCOTT

*Department of Statistical Science, Duke University,
Durham, North Carolina 27708-0251, U.S.A.
james@stat.duke.edu*

AND CARLOS M. CARVALHO

*Graduate School of Business, University of Chicago,
Chicago, Illinois 60637, U.S.A.
carlos.carvalho@chicagogsb.edu*

Original: September 2007

Revision: March 2008

ABSTRACT

We describe a serial algorithm called feature-inclusion stochastic search, or FINCS, that uses online estimates of edge-inclusion probabilities to guide Bayesian model determination in Gaussian graphical models. FINCS is compared to MCMC, to Metropolis-based search methods, and to the popular lasso; it is found to be superior along a variety of dimensions, leading to better sets of discovered models, greater speed and stability, and reasonable estimates of edge-inclusion probabilities. We illustrate FINCS on an example involving mutual-fund data, where we compare the model-averaged predictive performance of models discovered with FINCS to those discovered by competing methods.

Some key words: Covariance selection; Metropolis algorithm; lasso; Bayesian model selection; hyper-inverse Wishart distribution

1 INTRODUCTION

Gaussian graphical modeling offers a potent set of tools for shrinkage and regularization of covariance matrices in high-dimensional problems. Yet inferring the conditional independence structure of a random vector presents a substantial problem in stochastic computation. These model spaces are usually enormous; p nodes in a graph mean $m = p(p - 1)/2$ possible edges, and hence 2^m possible graphs corresponding to all combinations of individual edges being in or out of the model. Beyond $p = 7$, enumeration becomes a practical impossibility, and yet there are substantial gains to be had by fitting graphical models to far larger data sets—from portfolio selection problems with dozens or hundreds of assets, to biological applications involving thousands of genes. This motivates the development of accurate, scalable search methodologies that are capable of finding good models, or at least distinguishing the important edges from the irrelevant ones.

One obvious candidate is the reversible-jump MCMC approach of Giudici and Green (1999), which works well on very small problems—for example, four to six nodes. Yet many authors express skepticism that MCMC is well-suited for problems that are even slightly larger; see, for example, Dobra et al. (2004), Jones et al. (2005), and Hans et al. (2007). We share their concerns, and find little comfort in infinite-runtime guarantees when MCMC is deployed in discrete model spaces of such size and complexity. As these and many other authors note, assessing whether a Markov chain over a multimodal space has converged to a stationary distribution is devilishly tough, and theoretical results exist only for the smallest of problems (Woodard 2007). Even when state-of-the-art “mixing” tactics are used—simulated tempering, parallel chains, adaptive proposal distributions—apparent finite-time convergence can prove to be a mirage.

Two alternative classes of graphical model-selection procedures have arisen to sidestep these problems: compositional methods and direct search.

Compositional methods exploit the fact that graphs are models for conditional independence. They proceed by selecting a sparse regression model for each variable in terms of all the others, and then cobbling together the resulting set of conditional relationships into a graph to yield a valid joint distribution. Several methods for choosing each regression model are available, some based upon $L1$ -regularization (Meinshausen and Bühlmann 2006; Yuan and Lin 2007) and others based upon stepwise selection (Dobra et al. 2004).

Like MCMC, direct-search methods operate in the space of graphs rather than the space of conditional regressions. Unlike MCMC, however, they abandon the goal of converging to a stationary distribution—which may be unattainable, and is usually unassessable—in favor of simply listing and scoring a collection of good models. (Typically the scores are either Bayesian marginal likelihoods or posterior probabilities.) The two most prominent search procedures are the serial Metropolis-based algorithm of Jones et al. (2005) and the parallel Shotgun Stochastic Search of

Hans et al. (2007).

Our goal in this paper is to recommend an alternative search procedure that we call FINCS, for feature-inclusion stochastic search. FINCS is a serial procedure that relies upon constantly updated estimates of edge-inclusion probabilities in order to propose new regions of model space for exploration, and incorporates a novel, efficient global move based upon recent graph-triangulation algorithms. It is strongly related to the method recommended by Berger and Molina (2005) in the context of linear-model selection.

We first give background material on graphs and graphically constrained covariance matrices in §2. Then in §3 we describe the FINCS algorithm in detail. The remainder of the paper compares FINCS to other search procedures, to MCMC, and to compositional methods. These comparisons show FINCS to be superior on three objective criteria: stability, model scores, and predictive accuracy of models discovered. §4 and §5 describe these results on simulated problems at the lower (25-node) and upper (100-node) ends of what might be considered moderate-dimensional. Then in §6 we assess the predictive performance of models discovered using FINCS on a real 59-dimensional example involving monthly mutual-fund returns. We summarize our recommendations in §7.

2 BACKGROUND AND MODEL STRUCTURE

2.1 Notation and distributional theory for constrained covariance matrices

An undirected graph is a pair $G = (V, E)$ with vertex set V and edge set $E = \{(i, j)\}$ for some pairs $(i, j) \in V$. Nodes i and j are adjacent, or neighbors, if $(i, j) \in E$. Complete graphs are those having $(i, j) \in E$ for every $i, j \in V$. Complete subgraphs $C \subset V$ are called cliques; two cliques that overlap in a set S are said to have S as a separator.

A partition of a graph G into subgraphs (A, S, B) such that $V = A \cup B$, $S = A \cap B$ is complete, and any path from a node in A to a node in B goes through the separator S is called a decomposition. A sequence of subgraphs that cannot be decomposed further are the prime components of a graph; if every prime component is complete, the graph is said to be decomposable.

A Gaussian graphical model (Dempster 1972) uses such a graphical structure to define a set of pairwise conditional independence relationships on a p -dimensional normally distributed vector $x \sim N(0, \Sigma)$. With precision matrix $\Omega = \Sigma^{-1}$, elements x_i and x_j of the vector x are conditionally independent (given the neighboring variables of each) if and only if $\Omega_{ij} = 0$. If $G = (V, E)$ is an undirected graph representing the joint distribution of x , $\Omega_{ij} = 0$ for all pairs $(i, j) \notin E$. The covariance matrix Σ lives in $M(G)$, the set of all positive-definite matrices having elements in Σ^{-1} set to zero for all $(i, j) \notin E$.

The hyper-inverse Wishart distribution over a decomposable graph G is the unique

strong hyper-Markov distribution for $\Sigma \in M(G)$ with consistent clique marginals that are inverse Wishart distributed (Dawid and Lauritzen 1993). For a decomposable graph G , writing $(\Sigma | b, D, G) \sim \text{HIW}_G(b, D)$ means two things. First, it means that the density of Σ decomposes as a ratio of products over cliques and separators:

$$p(\Sigma | b, D, G) = \frac{\prod_{P \in \mathcal{P}} p(\Sigma_P | b, D_P)}{\prod_{S \in \mathcal{S}} p(\Sigma_S | b, D_S)}, \quad (1)$$

where Σ_C is the submatrix of Σ corresponding to the variables in clique C . This factorization holds if we assume, as in Dawid and Lauritzen (1993), that if $S = P_1 \cap P_2$ the elements of Σ_S are common in Σ_{P_1} and Σ_{P_2} .

Second, the $\text{HIW}(b, D)$ distribution means that for each clique C , $\Sigma_C \sim \text{IW}(b, D_C)$ with density:

$$p(\Sigma_C | b, D_C) \propto |\Sigma_C|^{-(b/2+|C|)} \exp \left\{ -\frac{1}{2} \text{tr} (\Sigma_C^{-1} D_C) \right\} \quad (2)$$

Note that, while this is the marginal distribution for each clique-specific block of Σ , there are still dependencies across different cliques induced by shared variables.

In nondecomposable graphs, the presence of additional component-level restrictions invalidates the closed-form density for Σ_C in (2). Computing marginal likelihoods for these structures requires approximating the normalizing constant of a non-conjugate distribution, which rapidly becomes too onerous a computational burden as dimension grows. Hence we restrict our attention to the much-simpler decomposable case and denote by \mathcal{G} the set of all decomposable graphs over a given number of nodes.

2.2 Marginal likelihoods for graphs

Let X be the matrix of vector-valued observations concatenated by row, and let $H(X'X)$ is the hyper-Markov sum-of-squares matrix corresponding to the prime components and separators of G . Using HIW priors for Σ , the marginal likelihood for G can be expressed as the ratio of the prior to the posterior normalizing constant:

$$p(X | G) = (2\pi)^{-np/2} \frac{h(G, b, D)}{h(G, b^*, D^*)} \quad (3)$$

where b is the prior degrees-of-freedom, D is the prior HIW location matrix, $b^* = b + n$, and $D^* = D + H(X'X)$. The normalizing constant of a hyper-inverse Wishart distribution is itself a function of the normalizing constants of its prime components and separators:

$$h(G, b, D) = \frac{\prod_{P \in \mathcal{P}} \left| \frac{1}{2} D_P \right|^{\frac{(b+|P|-1)}{2}} \Gamma_{|P|} \left(\frac{b+|P|-1}{2} \right)^{-1}}{\prod_{S \in \mathcal{S}} \left| \frac{1}{2} D_S \right|^{\frac{(b+|S|-1)}{2}} \Gamma_{|S|} \left(\frac{b+|S|-1}{2} \right)^{-1}} \quad (4)$$

where $\Gamma_p(x) = \pi^{p(p-1)/4} \cdot \prod_{j=1}^p \Gamma(x + (1-j)/2)$ is the multivariate gamma function.

The marginal likelihood of a graph is necessary to compute the posterior edge-inclusion probability for an edge (i, j) :

$$q_{ij} = \Pr(\Omega_{ij} \neq 0 \mid X) = \frac{\sum_G 1_{(i,j) \in G} \cdot P(X \mid G) \pi(G)}{\sum_G P(X \mid G) \pi(G)} \quad (5)$$

where $\pi(G)$ is the prior probability of the graph. These in turn are useful for defining the median probability graph: $G_M = (V, E_M)$, where $E_M = \{(i, j) : q_{ij} \geq 0.5\}$. This is by analogy with the median-probability linear model of Barbieri and Berger (2004).

In this paper, we compute marginal likelihoods using the fractional-Bayes approach of Carvalho and Scott (2007):

$$p(X \mid G) \equiv Q(X, g \mid G) = (2\pi)^{-np/2} \frac{h(G, gn, gH(X'X))}{h(G, n, H(X'X))} \quad (6)$$

with $h(G, b, D)$ defined as in (4), and for some suitable choice of g that is $O(n^{-1})$. These marginal likelihoods are akin to using a set of g -priors (Zellner 1986) for doing variable selection on each univariate conditional regression, and have a number of desirable properties relating to the notion of information consistency, as defined by Liang et al. (2007).

The other possibility for computing marginal likelihoods is to use the conventional shrinkage prior found in, for example, Giudici and Green (1999), Dobra et al. (2004) and Jones et al. (2005). The conventional prior sets $\Sigma \sim \text{HIW}(\delta, \tau I)$, with $\delta = 3$ giving Σ a finite first moment, and with τ chosen to match the marginal variance of the data in order to provide an appropriate scale for the prior (which is necessary to get reasonable results in model-selection problems). This prior does give closed-form marginal likelihoods, and indeed has been the prior of choice in many previous studies due to the lack of a reasonable alternative.

Yet these priors tend to perform very poorly in contrast to the fractional prior, leading to an unintuitively high level of uncertainty and an artificial flattening of modes in model space. As Carvalho and Scott (2007) show, this happens because the conventional prior induces a set of ridge-regression priors on each univariate conditional regression, in contrast to the induced g -like priors of the fractional-Bayes approach. Zellner and Siow (1980), among others, describe why ridge-regression priors are suboptimal for variable selection, suggesting the source of the problem for the conventional $\text{HIW}(\delta, \tau I)$ prior in this context.

For this reason we primarily use the fractional prior, with the understanding that its sharper characterization of model uncertainty will allow a fairer comparison of

different computing strategies. Note that g indicates the fraction of the likelihood used in training a set of noninformative priors on $\{\Sigma_G : G \in \mathcal{G}\}$. Here, we take the default value $g = 1/n$, implying that a priori, the vector x has a marginal hyper-Cauchy distribution (Dawid and Lauritzen 1993). More details on fractional Bayes factors can be found in O’Hagan (1995).

2.3 Multiplicity-correction priors over graphs

Covariance selection is an implicit problem in multiple hypothesis testing, where each null hypothesis corresponds to the exclusion of a single edge from the graph. This motivates an edge-selection prior of the form:

$$\pi(G) = r^k(1 - r)^{m-k} \tag{7}$$

for a graph G having k edges out of m possible ones. Note that if r is treated as a model parameter to be estimated from the data, then this prior has an automatic adjustment for multiple testing as the number of possible edges grows (Carvalho and Scott 2007; Scott and Berger 2008). The resulting model probabilities yield very strong control over the number of false edges admitted into the model, which can be shown to remain bounded even as the number of spurious tests grows without bound.

For choices of the prior distribution $\pi(r)$ in the beta family, r can be explicitly marginalized out to induce a set of prior model probabilities that will automatically create the right multiplicity-correction behavior; see also Scott and Berger (2006). The default uniform prior on r gives a marginal prior inclusion probability of $1/2$ for all edges and yields model probabilities of the form:

$$\pi(G) = \frac{(k)!(m - k)!}{(m + 1)(m!)} = \frac{1}{m + 1} \binom{m}{k}^{-1} \tag{8}$$

for G have k edges out of m possible ones.

We refer to this as fully Bayesian multiplicity correction to distinguish it from the empirical-Bayes ideas explored in, for example, George and Foster (2000) and Cui and George (2007) in the context of regression variable selection. We prefer the fully Bayesian approach due to the issues with empirical Bayes discussed in Scott and Berger (2008), and we use these priors throughout.

3 GRAPHICAL MODEL DETERMINATION

3.1 Existing methods

3.1.1 Markov-chain Monte Carlo

Giudici and Green (1999) popularized the use of MCMC for graphical models, implementing a reversible-jump sampler (Green 1995) over all model parameters including

graphical constraints.

Jones et al. (2005) then considered a version of MCMC that eliminated the complexities of reversible-jump by explicitly marginalizing over most parameters and working directly with graph marginal likelihoods. This version of the algorithm makes one-edge moves through graph space, accepting proposed moves with probability:

$$\alpha = \min \left(1, \frac{p(G')h(G | G')}{p(G)h(G' | G)} \right) \quad (9)$$

where $p(\cdot)$ is the posterior probability of the graph (available in closed form due to conjugacy assumptions), and $h(\cdot)$ is the proposal probability.

Another obvious MCMC algorithm to apply to graphical models is Gibbs sampling, whereby each edge indicator variable in turn is sampled conditional upon all the others. This is the graphical-model equivalent of the SSVS procedure described by George and McCulloch (1993) for linear models, and we might call it stochastic-search edge selection. Yet this algorithm, despite being a true MCMC with demonstrable theoretical convergence to a stationary distribution, does not receive any attention in the literature. As we show on an example below, it often completely misses very large modes in model space, which explains why people seem not to trust it.

3.1.2 Metropolis-based stochastic search

In practice, the Metropolis criterion is less useful as an MCMC transition kernel, and far more useful as a search heuristic for finding and cataloguing good models. This is true for two reasons.

For one thing, guaranteeing ergodicity of the Markov chain requires that each step involve a costly enumeration of all possible one-edge moves that maintain decomposability (which in general will not be symmetric). If this fact is not accounted for, then the proposal densities $h(\cdot)$ in (9) will be wrong, and the chain will not converge to a stationary distribution. This enumeration is possible, but costly (Deshpande et al. 2001).

The more important reason, however, is the lack of MCMC convergence diagnostics on such complex, multimodal problems. The model spaces are simply too large to trust the usual rules of thumb, and it makes little sense to evaluate models by their frequency of occurrence in a Monte Carlo sample when we can simply list the best ones instead. Hence it is best to view the Metropolis algorithm as a tool for stochastic search, and not as true MCMC.

We note one recent advancement, also described in Jones et al. (2005) and developed further by Hans et al. (2007), called Shotgun Stochastic Search (SSS). It powerfully exploits a distributed computing environment to consider all possible local moves in parallel at each step, moving to a new model in proportion to how much each possible move improves upon the current model. Yet for those who only have

access to serial computing environments, evaluating all possible neighbors of a given graph may be prohibitively time-consuming.

3.1.3 Compositional methods

In compositional search, one first defines the neighborhood $\text{ne}(i)$ of each node by fitting a single sparse regression model of x_i upon a subset of $\{x_j : j \neq i\}$. Dobra et al. (2004) perform a Bayesian selection procedure to get each neighborhood, while Meinshausen and Bühlmann (2006) and Yuan and Lin (2007) use variants of the lasso (Tibshirani 1996). Regardless of the variable-selection method used, the resulting set of regressions implicitly defines a graph.

Such procedures do not, in general, yield a valid joint distribution: often $i \in \text{ne}(j)$ but $j \notin \text{ne}(i)$, which is impossible in an undirected graph. There are two ways of proceeding to a valid edge set E , which we call (for obvious reasons) the AND graph and the OR graph:

AND graph: $(i, j) \in E$ if $i \in \text{ne}(j) \wedge j \in \text{ne}(i)$

OR graph: $(i, j) \in E$ if $i \in \text{ne}(j) \vee j \in \text{ne}(i)$

Each edge set must then be triangulated to yield decomposable graphs. It is easy to see that $E_\wedge \subset E_\vee$. There is no principled way to decide which edge set to use in practice, though Meinshausen and Bühlmann (2006) give conditions under which the two will converge to the same answer asymptotically.

Note that defining a graph by composition also involves a (possibly difficult) search procedure, since each sparse regression model must be chosen from 2^p possible candidates. Lasso does not involve an explicit search over models, but the $L1$ penalty term that induces sparsity is typically chosen by cross-validation, which often takes just as long when done carefully.

3.2 FINCS: feature-inclusion stochastic search

As an alternative, we present a serial algorithm called FINCS, or feature-inclusion stochastic search, that combines three types of moves through graph space: local moves, resampling moves, and global moves. Related to simpler algorithm introduced by Berger and Molina (2005) in the context of regression variable selection, FINCS is motivated by a simple observation: edge moves that have improved some models are more likely to improve other models as well, or at least more likely to do so than a randomly chosen move.

FINCS also recognizes the tension between two conflicting goals: local efficiency and global mode-finding. Results from Giudici and Green (1999) and Wong et al. (2003) suggest that pairwise comparisons for edge-at-a-time moves in graph space are much faster than those for multiple-edge moves due to the local structure implied by

the hyper-inverse Wishart distribution. Both a graph’s junction tree representation and its marginal log-likelihood can be updated quite efficiently under such local moves, which will affect at most two cliques and one separator.

Yet we must also confront the familiar problem of multimodality, severely exacerbated by the restriction to decomposable graphs. Each local move changes not only the graph itself but also the local topology of reachable space, opening some doors to immediate exploration and closing others. The problem only gets worse as the number of nodes increases, since the stepwise-decomposable paths in model space between two far-flung graphs become vanishingly small as a proportion of all possible paths between them. The theory guarantees that such a path always exists (Frydenberg and Lauritzen 1989), but it may be very difficult to find.

These principles suggest that a sound computational strategy must include a blend of local and global moves—local moves to explore concentrated regions of good graphs with minimal numerical overhead, and global moves to avoid missing important regions that aren’t easily reachable from one another by a series of local moves that maintain stepwise decomposability.

Motivated by these concerns, FINCS interleaves three different kinds of moves:

Local move: Starting with graph G_{t-1} , generate a new graph G_t by randomly choosing to add or delete an edge that will maintain decomposability. If adding, do so in proportion to \hat{q}_{ij} , the current estimates of edge-inclusion probabilities. If deleting, do so in inverse proportion to these probabilities.

Resampling move: Revisit one of $\{G_1, G_2, \dots, G_{t-1}\}$ in proportion to their posterior model probabilities, and begin making local moves from the resampled graph.

Global move: Jump to a new region of graph space by generating a *randomized median triangulation pair*, or RMTP. This can be done in three steps:

1. Begin with an empty graph and iterate through all possible edges once, independently adding each one in proportion to its estimated inclusion probability \hat{q}_{ij} . In general this will yield a nondecomposable graph G_N . A simpler variation involves deterministically choosing the median graph.
2. Compute a minimally sandwiching triangulation pair for G_N . This pair comprises a minimal decomposable supergraph $G^+ \supset G_N$ along with a maximal decomposable subgraph $G^- \subset G_N$, wherein no candidate edge can be added to G^- or removed from G^+ while still maintaining the decomposability of each.
3. Evaluate each member of the pair, and choose G_t in proportion to their posterior probabilities.

An RMTP move immediately transcends the limitations of stepwise-decomposable moves, bridging nondecomposable valleys in model space with a minimal set of fill

edges and allowing the search to escape local modes. Several algorithms are available for computing these inclusion-minimal triangulations and inclusion-maximal subtriangulations in $O(nk)$ time, where n is the number of nodes and k is the number of edges in G_N . One nice algorithm due to Berry et al. (2006) allows simultaneous computing of both G^+ and G^- ; another can be found in Berry et al. (2004). It should be noted that these triangulations are neither unique nor globally optimal, since computing a minimum triangulation is a different (NP-complete) problem.

After each step, simply compute the posterior probability of G_t , and use it (assuming it hasn't been visited already) to update the estimated inclusion probabilities of all the edges:

$$\hat{q}_{ij}(t) = \frac{\sum_{k=1}^{k=t} \mathbf{1}_{(i,j) \in G_k} \cdot P(X | G_k) \cdot \pi(G_k)}{\sum_{k=1}^{k=t} P(X | G_k) \cdot \pi(G_k)} \quad (10)$$

It is important to emphasize that these inclusion probabilities \hat{q}_{ij} are simply a search heuristic. There is no sense in which they “converge” to the true inclusion probabilities, except in the trivial sense that FINCS will eventually enumerate all the models. We recognize that present theory and computing technology simply do not allow us to make any definitive statements about the true inclusion probabilities in (5), and that the only unambiguous measure of a search procedure is: which models did it find, and how good are they?

We do believe, of course, that the estimated inclusion probabilities provide a useful summary of the search, since one can tell just by glancing at them how important each edge seems to be among the cohort of models discovered. They are also nice for judging the richness of an estimate, in that lots of 1's and 0's indicate a very top-heavy list of models. And they are great for assessing stability, since it would be foolish to trust a procedure that yields highly volatile estimates under repetition.

3.3 Details of data structures and implementation

FINCS, Metropolis, and Gibbs sampling were implemented in C++ on a 3.4 GHz Dell Optiplex PC running Linux. Our object-oriented code makes extensive use of the Standard Template Library (STL) for representing the complicated data structures required by graphical models, and uses the Newmat C++ matrix library for matrix operations.

Substantial gains in efficiency for the resampling step are possible by implementing storage of previously visited models in a binary search tree (a map in the C++ STL) indexed by model score. This requires normalizing model probabilities to $(0, 1]$ and maintaining a lower-dimensional representation of the empirical distribution of model scores on that interval. We use a beta distribution for this purpose, whose parameters are updated each time a substantial pocket of probability is found in model space. By drawing a score from this beta distribution and then resampling the model corresponding to that score, resampling can be done in $\log T$ time, where T is

the current number of saved models. This allows only approximate resampling, but experience suggests that it is far preferable to the costly linear-time step required by exact resampling (which is not, after all, a theoretical requirement of the algorithm).

Subsequent local moves can also be greatly streamlined by saving a copy of the junction tree for each graph so that it doesn't need to be recomputed upon resampling.

In our experience, a blend of 80–90% local moves with 10-15% resampling moves seems to work well, with the remaining fraction devoted to global moves; these are most effective when used sparingly due to the computational expense of triangulating the graph and rebuilding the junction tree, which grows quite rapidly with dimension. It also seems advisable to allow for an unusually long run of local moves following a global move. This accounts for the fact that our global move can be expected to find other hills in the model space, but is very unlikely to jump directly to the tops of those hills. A longer-than-normal run of local moves following such a jump allows the search procedure to climb to the top, thereby helping to solve the multimodality problem.

In larger problems, we also recommend initializing the search with a cohort of promising graphs for resampling, which can be done automatically and quite rapidly before beginning the search. We have found that on small-to-moderate-dimensional problems such as the 25-node example in Section 4, this step can safely be skipped, since FINCS converges quite rapidly to the same answer regardless of the initial graph.

Finally, it is necessary to bound the inclusion probabilities away from 0 and 1 in order to allow new edges to enter the picture. This is done by renormalizing the online estimates for probabilities to $(\delta, 1 - \delta)$ for some suitably chosen small value. Different choices of δ will either flatten or sharpen the choice of edges; a reasonable default choice in moderate-dimensional problems is between 0.01 and 0.05.

A C++ package implementing FINCS is available from the authors upon request.

4 A SIMULATED 25-NODE EXAMPLE

We begin by comparing FINCS with both Metropolis and Gibbs on the 25-node graph in Figure 1. Only the first 10 nodes have any connectivity structure and so the remaining 15 are omitted from the diagram. This problem foregrounds two challenges: to find the smaller 10-node graph embedded in the larger 25-node space, and to avoid flagging false edges that follow from getting stuck in local modes.

The results of these comparisons are summarized in Figures 2, 3, and 4. Here we use the global-move version of FINCS, which resamples an old model every 10 iterations and makes an RMTP move every 50 iterations. “Gibbs” refers to stochastic-search edge-selection, modeled after George and McCulloch (1993); “Metropolis” refers to the random-walk stochastic-search algorithm of Jones et al. (2005).

Three questions are of interest:

1. Which search method finds the best collection of models, as measured by posterior probability?

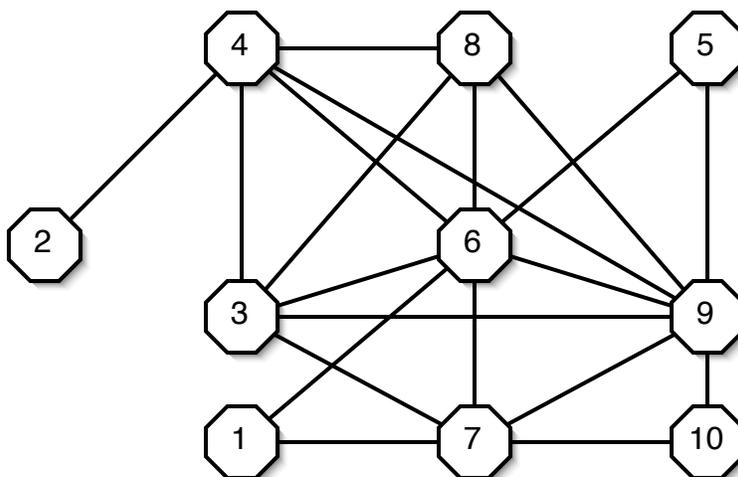


Figure 1: True graph for the 25-node example. Since nodes 11 through 25 are unconnected, both to this subgraph and to each other, they are omitted for visual simplicity.

2. Which search procedures are stable under repetition?
3. Are the estimated inclusion probabilities from FINCS and Metropolis intuitively reasonable?

Figure 2 gives histograms of the best 1000 models discovered during single long runs of FINCS, Gibbs, and Metropolis. This decisively answers the first question: all of the top 1000 models discovered by FINCS are more probable than the single best model discovered by either Metropolis or Gibbs. Gibbs does particularly poorly here; its best model was 11 orders of magnitude worse than the thousandth-best model discovered by FINCS. Such large systematic differences were unexpected, and yet were very stable under repeated restarts, both from the same initial model and very different initial models. On this example, FINCS always found better models than both Metropolis and Gibbs, and there were always substantial gaps between the “worst of the best” discovered by FINCS and the best discovered by either competing method.

To formally assess stability, we conducted 20 short runs of each algorithm, starting from the null graph each time. (The run lengths were 2000, 300000, and 1 million for Gibbs, FINCS, and Metropolis, respectively.) Each run yielded an estimate for all 300 pairwise inclusion probabilities. The standard errors of these estimates provide an excellent proxy for stability, since highly variable inclusion probabilities indicate that the results of a single run are untrustworthy.

Figure 3 shows the results of this exercise. The estimated inclusion probabilities for the Metropolis algorithm are extremely unstable, displaying run-to-run standard deviations as high as 40% for some edges. Indeed, these edges were estimated at

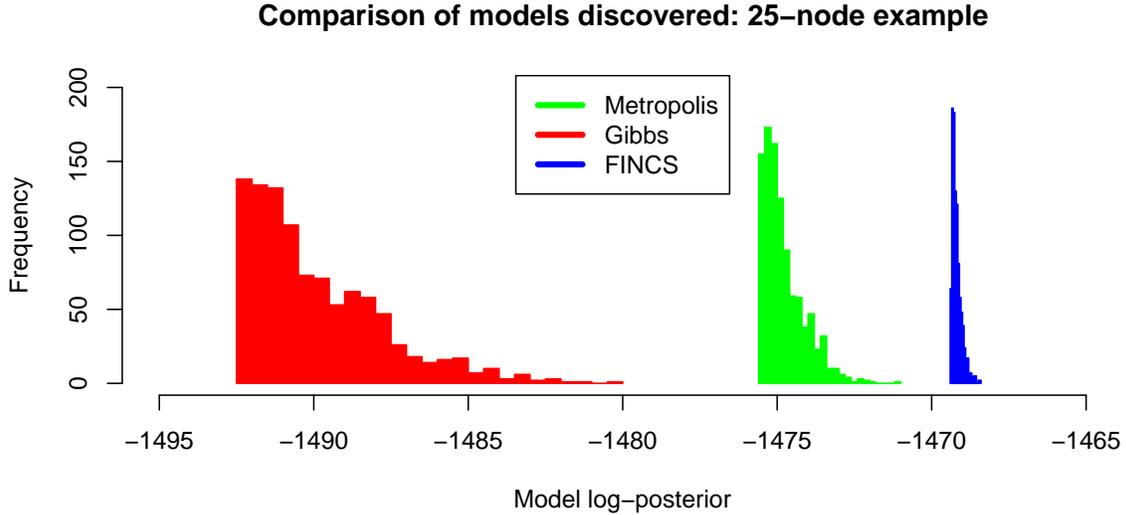


Figure 2: Comparison of model-search algorithms on the 25-node example. These are the posterior probabilities of the top 1000 models found using Gibbs (10000 iterations), FINCS (3 million iterations), and Metropolis (10 million iterations). The Gibbs and FINCS runs each had the same number of marginal likelihood evaluations.

0% inclusion probability in some runs and 100% in others, which is very unsettling behavior. Gibbs and FINCS, on the other hand, were fairly stable, with estimated inclusion probabilities rarely differing by more than 5% from run to run.

Finally, Figure 4 gives the estimated edge-inclusion probabilities for long runs of all three algorithms. As the above results foreshadow, Metropolis did quite poorly: it flagged several false positives outside the 10-node subgraph, and missed many edges corresponding to strong partial correlations. We conclude that Metropolis is uniformly dominated by FINCS for problems of this size, since FINCS finds better models and is far more stable.

While it is easy to say that Metropolis gets the inclusion probabilities wrong, it is more difficult to say whether FINCS gets them right. For both Gibbs and FINCS, there is very strong rank-correlation ($> 90\%$) between pairwise inclusion probabilities and pairwise partial correlations, which suggests that both procedures are flagging important edges. And as Figure 4 shows, FINCS does yield inclusion probabilities that are fairly close to those given by Gibbs, with slight-but-systematic biases in favor of strong edges and against weak ones (as might be expected, given the “expand about the modes” nature of the procedure). We view these biases as rather minor given the enormous size of the model space—indeed, both procedures give the same median-probability graph, and yield the same qualitative conclusions.

The Gibbs inclusion probabilities do, of course, represent a true MCMC estimate,

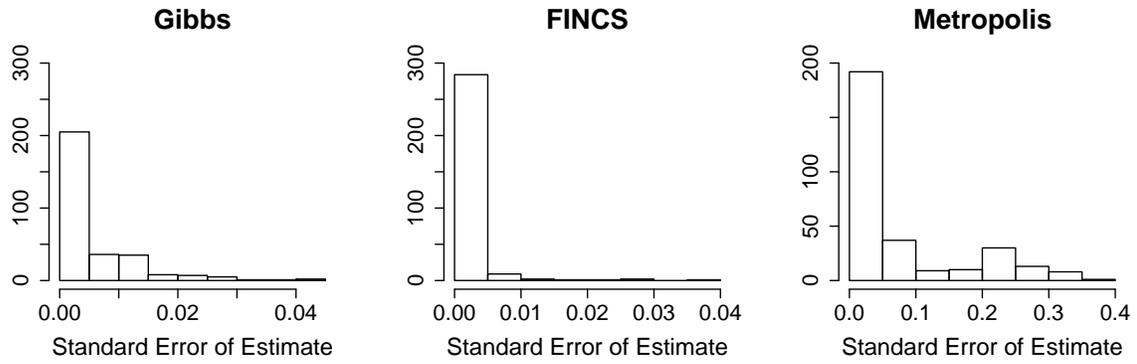


Figure 3: Standard errors (under 20 repetitions) of estimates for the 300 pairwise inclusion probabilities in the 25-node example.

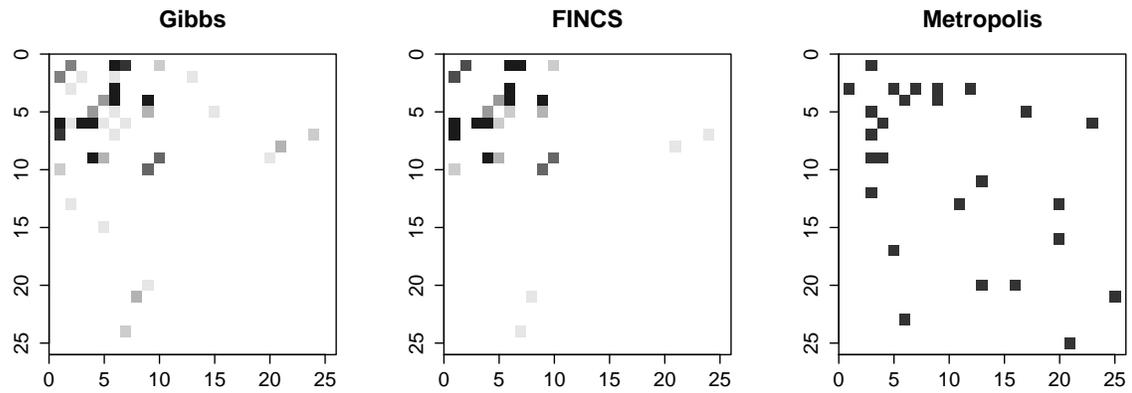


Figure 4: Grayscale images of the estimated inclusion probabilities from each algorithm on the 25-node example.

and they meet the usual informal convergence criterion: after a very brief burn-in, they are substantially the same run after run, regardless of the starting point. Readers willing to trust the Gibbs estimates, therefore, are likely to conclude that FINCS gets things slightly wrong here (though not nearly so wrong as Metropolis).

We are leery of trusting the Gibbs estimates, however, because of the evidence in Figure 2: the Gibbs inclusion probabilities, despite being highly stable under repetition, are estimated without ever visiting a single model within 11 orders of magnitude of the best model found by FINCS. This suggests that the usual convergence diagnostics may be misleading. This is uncertain; it is certain, however, that the Gibbs sampler routinely misses enormous pockets of probability in model space, and produces a demonstrably inferior list of models.

In the absence of better theory or more trustworthy convergence diagnostics, it seems imprudent to use the Gibbs answer when FINCS finds thousands of models that are each tens of thousands of times more probable than any found by Gibbs. At worst, FINCS is turning up the most likely models while giving a useful, albeit slightly biased, picture of the inclusion probabilities. At best, it is the only algorithm providing summaries worth trusting, since at least we know what we are getting when we list the best models we can find.

5 A SIMULATED 100-NODE EXAMPLE

To assess the performance of FINCS on a bigger problem, we simulated a data set of size 250 from the correlation matrix of a stationary AR(10) process of length 100. This represents a graphical model due to the block-diagonal form of the 100-dimensional precision matrix. We then ran FINCS-global, FINCS-local, and Metropolis 20 different times, always starting from the null graph, and recorded the top marginal log-likelihood discovered in the course of the search. Here, FINCS-local resamples a previously visited model every 5 iterations. FINCS-global resamples every 5 iterations and performs a global move every 1000 iterations; each global move is followed by 100 local moves as described in §3.3. Results are presented in the Figure 5, while runtime information is in Table 1.

To give a better sense of time efficiency, we performed two pairs of approximately equal-time, moderate-length runs of Metropolis and FINCS-global on the same data set. One pair of runs started from the null graph with no edges, while another pair started from an initial graph corresponding to a set of conditional regressions. Table 2 gives the top marginal log-likelihoods discovered, along with runtime information for each search.

There are four lessons to take from these experiments.

First, our proposed global move is very helpful for escaping local modes in model space. The overhead required to triangulate and compute a new junction tree means that FINCS-global takes about twice as long as Metropolis for an equivalent number of steps, implying that a single global move takes roughly the same amount of time as

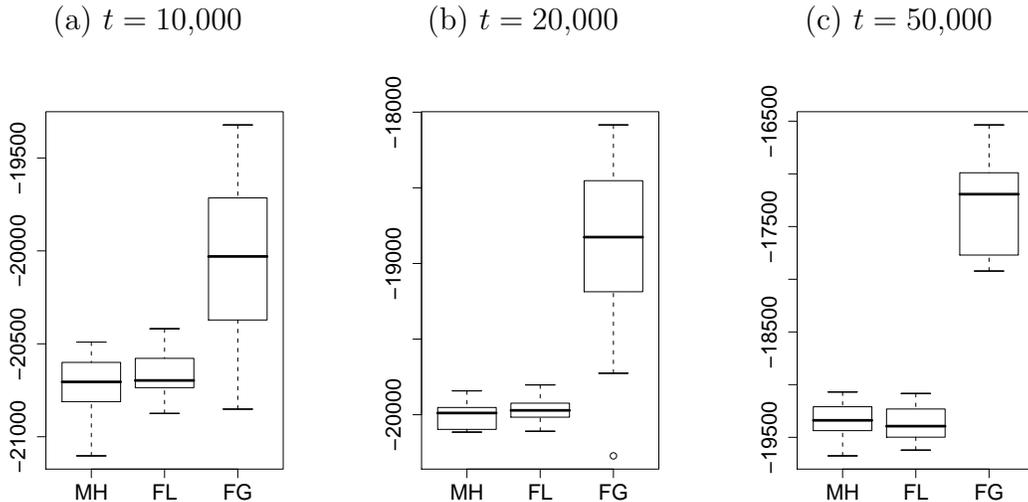


Figure 5: 100-node example: Boxplots of the top marginal log-likelihoods discovered (y -axis) on 20 restarts of 3 different run lengths for the Metropolis (MH), FINCS-local (FL), and FINCS-global (FG) algorithms.

Iterations	Runtime in seconds		
	Metropolis	FINCS-local	FINCS-global
10,000	20.98 (0.58)	27.22 (1.00)	36.36 (2.96)
20,000	38.25 (1.08)	62.86 (3.53)	68.03 (5.29)
50,000	99.28 (2.63)	155.38 (5.12)	176.21 (14.42)

Table 1: 100-node example: Mean (standard deviation) runtime for each algorithm.

Algorithm	Start	Iterations	Time (s)	Best model
FINCS-global	Null graph	80,000	458.90	-16897.32
Metropolis	Null graph	220,000	456.70	-18689.34
FINCS-global	Guess ($ z > 3.0$)	50,000	318.74	-3211.09
Metropolis	Guess ($ z > 3.0$)	110,000	334.72	-3330.84

Table 2: Marginal log-likelihoods of best models discovered from two different starting points. The guessed graph corresponds to a set of conditional regressions (marginal log-likelihood of -5170.29 after triangulation), where an edge (i, j) was flagged if x_j yielded a z -statistic ≥ 3.0 in absolute value as a predictor of x_i (or vice versa).

1000 local moves for this 100-node problem. (This ratio gets steeper as the number of nodes increases). Yet the advantage conferred by this global move is clear; 10,000 iterations of FINCS-global, for example, dramatically outperformed 20,000 iterations of Metropolis despite taking about 25% less raw time on average.

The necessity of the global move becomes even clearer as we take more iterations ($t = 50,000$ in the right panel of Figure 5). By 50,000 iterations, both Metropolis and FINCS-local have leveled off as a result of getting stuck in local modes, whereas FINCS-global has continued to climb at a rapid pace.

Second, there are stark differences in character between the posterior summaries yielded by Metropolis and by FINCS. Metropolis acts essentially as a stochastic optimizer; in an average run of 10,000 iterations, it visited fewer than 300 distinct models. FINCS, on the other, both finds modes and expands about them, cataloguing many thousands of distinct models (including hundreds that are nearly as good as best model found) in each run of 10,000 iterations. This yields a much richer summary of model space, more than compensating for the marginal time penalty paid to store and resample models.

Third, no search algorithm, no matter how well tuned, can compensate for a poor starting point on problems of this size. Even FINCS-global, despite being a very adept hill-climber compared to other competing algorithms, takes a very long time to bridge the many thousands of orders of magnitude between the unreasonable null graph and a reasonable guess. This is very different from our 25-node problem, where the choice of initialization did not matter at all.

Finally, even when a smart initial guess is supplied, FINCS still substantially outperforms Metropolis. Given the same amount of computing time, FINCS found models 119 orders of magnitude better than those found by Metropolis. While this looks small compared to the 2000 orders of magnitude separating the best models that each procedure found in an uninitialized search, it still corresponds to a Bayes factor of 4.8×10^{51} —an enormous improvement.

6 A REAL 59-NODE EXAMPLE: MUTUAL FUNDS

We now give a real 59-node example involving mutual-fund data; our goal is to compare FINCS and Metropolis both to each other and to the lasso, a popular compositional method.

Our example involves graphical-model selection for a set of 59 mutual funds in several different sectors: 13 U.S. bond funds, 30 U.S. stock funds, 7 balanced funds investing in both U.S. stocks and bonds, and 9 international stock funds. The example is motivated by the need for accurate, stable estimates of variances and pairwise correlations of assets in dynamic portfolio-selection problems. Graphical models, as Carvalho and West (2007) show, offer a potent tool for regularization and stabilization of these estimates, leading to portfolios with the potential to uniformly dominate their traditional counterparts in terms of risk, transaction costs, and overall profitability.

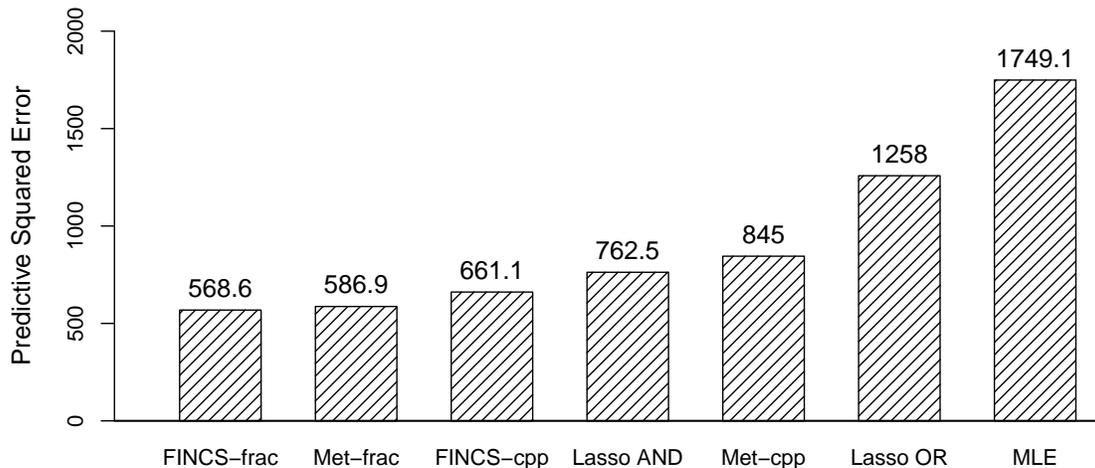


Figure 6: Sum of squared errors in predicting missing values, 59-node mutual-fund example. “Frac” refers to fractional marginal likelihoods, while “cpp” refers to marginal likelihoods under the conventional proper prior. We allowed 8 million iterations for Metropolis and 3 million for FINCS. Runtimes were: FINCS-frac (12m, 21s); FINCS-cpp (13m, 1s); Met-frac (27m, 46s); Met-cpp (28m, 31s).

A fair barometer of performance here is prediction, since we are including a non-Bayesian technique (the lasso) for which there is no notion of the marginal likelihoods and posterior probabilities used as measures in previous sections.

We split the 86-month sample into a 60-month training set (since we wish to compare graphical estimates to the unrestricted 59-dimensional covariance matrix) and a 26-month prediction set. We then used the training set to search for good models (using both FINCS-global and Metropolis) and compute posterior means $\{\hat{\Sigma}\}$ under each of the 500 best models found during the course of the search. We then used these estimates to predict observations in the 26-month validation set. In each month, we used the 56 “observed” returns along with the estimated $\hat{\Sigma}$ ’s to compute the model-averaged conditional expectations of the remaining 3 “missing” returns (which are known in reality). (The numbers 56 and 3 were chosen because we can enumerate all $\binom{59}{3} = 32509$ combinations of 3 unobserved assets, thereby eliminating the possibility of error due to sampling.)

We performed these imputations using both search procedures along with two different methods of computing graph marginal likelihoods: the fractional approach used on the previous examples, and the conventional $HIW(\delta, \tau I)$ described in §2.2. We then computed the lasso solutions using leave-one-out cross-validation to choose the $L1$ penalty term, an oft-recommended procedure in the penalized-likelihood literature. We include the predictions of both the lasso-AND graph and the lasso-OR graph,

along with those of the MLE for the sake of comparison.

The total squared-errors of these model-averaged imputations are given in Figure 6. These results show that FINCS leads to better predictions in less computing time than Metropolis, confirming the trends seen on simulated data and bolstering our conclusion that FINCS is a better default algorithm for characterizing model uncertainty.

Using fractional priors, we had to run Metropolis over twice as long as FINCS in order to achieve a comparable sum of squared errors. An even larger discrepancy between algorithms appears under the HIW($\delta, \tau I$) prior, where Metropolis does 28% worse than FINCS in mean-squared error despite running over twice as long. The known adverse “mode-flattening” effect of the conventional prior (Carvalho and Scott 2007) appears to hamstring the Metropolis algorithm far more than it does FINCS—a gap which narrows, but does not close entirely, under the well-behaved fractional prior.

It is also interesting that three of the four Bayesian estimates beat the lasso-AND graph, with the best providing a 25% reduction in mean-squared error. All four Bayesian models beat the lasso-OR graph quite considerably—even the Bayesian procedure using the inferior prior and search algorithm. The two lasso graphs are also very different from each other. The asymptotic guarantee that the two will converge to the same answer is not especially helpful here. Without useful guidelines for choosing between them, one could easily be hamstrung—i.e. lose money—by choosing the vastly inferior lasso-OR graph *ex ante*.

7 DISCUSSION

We have introduced a stochastic-search algorithm based upon a novel approach to computation in Gaussian graphical models: using online estimates of inclusion probabilities both to drive the choice of models to visit, and to guide a new form of global move in graph space that allows escape from the local modes that tend to thwart other procedures. Simulations suggest that FINCS outperforms Metropolis regardless of the problem size, and regardless of the assessment metric. We have given general guidelines for setting the frequency of resampling and global moves, but on large problems, some ad-hoc tuning of these parameters, together with multiple restarts, may be necessary to yield optimal performance.

We also find FINCS, and search procedures in general, to be more trustworthy in this context than a pure MCMC method. Despite apparent convergence, Gibbs sampling has a worrisome tendency to miss large, important parts of model space, calling its utility here into question.

FINCS is a serial algorithm, yet gives reasonable answers on moderate-dimensional problems that up to now have required parallel methods such as Shotgun Stochastic Search. It therefore provides a crucial bridge between small problems for which Metropolis is clearly adequate, and large problems for which no serial algorithm will

be competitive. It seems particularly well-suited to problems like the 59-node mutual-fund example of Section 6, where the predictive context naturally calls for Bayesian model averaging. In this context, FINCS gives a demonstrably better cohort of models in substantially less time than Metropolis, and is not nearly as susceptible to the mode-flattening effect of a suboptimal model-selection prior on the covariance matrix.

FINCS also finds Bayesian models yielding much better predictions than the lasso, a popular classical method often specifically lauded for its predictive optimality.

REFERENCES

- Barbieri, M. and Berger, J. O. (2004), “Optimal predictive model selection,” *The Annals of Statistics*, 32, 870–897.
- Berger, J. O. and Molina, G. (2005), “Posterior model probabilities via path-based pairwise priors,” *Statistica Neerlandica*.
- Berry, A., Blair, J., Heggernes, P., and Peyton, B. (2004), “Maximum Cardinality Search for Computing Minimal Triangulations of Graphs,” *Algorithmica*, 39, 287–298.
- Berry, A., Heggernes, P., and Villander, Y. (2006), “A vertex incremental approach for maintaining chordality,” *Discrete Mathematics*, 306, 318–336.
- Carvalho, C. and West, M. (2007), “Dynamic Matrix-Variate Graphical Models,” *Bayesian Analysis*, 2, 69–96.
- Carvalho, C. M. and Scott, J. G. (2007), “Objective Bayesian model selection in Gaussian graphical models,” Tech. rep., Institute of Statistics and Decision Sciences.
- Cui, W. and George, E. I. (2007), “Empirical Bayes vs. fully Bayes variable selection,” *Journal of Statistical Planning and Inference*, to appear.
- Dawid, A. P. and Lauritzen, S. L. (1993), “Hyper-Markov laws in the statistical analysis of decomposable graphical models,” *The Annals of Statistics*, 3, 1272–1317.
- Dempster, A. (1972), “Covariance selection,” *Biometrics*, 28, 157–175.
- Deshpande, A., Garofalakis, M. N., and Jordan, M. I. (2001), “Efficient stepwise selection in decomposable models,” in *Uncertainty in Artificial Intelligence (UAI), Proceedings of the Seventeenth Conference*, eds. Breese, J. and Koller, D.

- Dobra, A., Jones, B., Hans, C., Nevins, J., and West, M. (2004), “Sparse graphical models for exploring gene expression data,” *Journal of Multivariate Analysis*, 90, 196–212.
- Frydenberg, M. and Lauritzen, S. L. (1989), “Decomposition of maximum likelihood in mixed models,” *Biometrika*, 76, 539–555.
- George, E. I. and Foster, D. P. (2000), “Calibration and empirical Bayes variable selection,” *Biometrika*, 87, 731–747.
- George, E. I. and McCulloch, R. (1993), “Variable selection via Gibbs sampling,” *Journal of the American Statistical Association*, 88, 881–889.
- Giudici, P. and Green, P. J. (1999), “Decomposable graphical Gaussian model determination,” *Biometrika*, 86, 785–801.
- Green, P. J. (1995), “Reversible jump Markov chain Monte Carlo computation and Bayesian model determination,” *Biometrika*, 82, 711–732.
- Hans, C., Dobra, A., and West, M. (2007), “Shotgun stochastic search in regression with many predictors,” *Journal of the American Statistical Association*, 102, 507–516.
- Jones, B., Carvalho, C., Dobra, A., Hans, C., Carter, C., and West, M. (2005), “Experiments in Stochastic Computation for High-dimensional Graphical Models,” *Statistical Science*, 20, 388–400.
- Liang, F., Paulo, R., Molina, G., Clyde, M., and Berger, J. (2007), “Mixtures of g -priors for Bayesian variable selection,” *Journal of the American Statistical Association*, to appear.
- Meinshausen, N. and Bühlmann, P. (2006), “High dimensional graphs and variable selection with the Lasso,” *Annals of Statistics*, 34, 1436–1462.
- O’Hagan, A. (1995), “Fractional Bayes factors for model comparison,” *Journal of the Royal Statistical Society, Series B*, 57, 99–138.
- Scott, J. G. and Berger, J. O. (2006), “An exploration of aspects of Bayesian multiple testing,” *Journal of Statistical Planning and Inference*, 136, 2144–2162.
- (2008), “Multiple Testing in Variable Selection: Empirical Bayes and the Information Gap,” Tech. rep., Duke University Department of Statistical Science.
- Tibshirani, R. (1996), “Regression shrinkage and selection via the lasso,” *J. Royal. Statist. Soc B.*, 58, 267–88.

- Wong, F., Carter, C., and Kohn, R. (2003), “Efficient estimation of covariance selection models,” *Biometrika*, 90, 809–830.
- Woodard, D. (2007), “Conditions for Rapid and Torpid Mixing of Parallel and Simulated Tempering on Multimodal Distributions,” Ph.D. thesis, Duke University.
- Yuan, M. and Lin, Y. (2007), “Model selection and estimation in the Gaussian graphical model,” *Biometrika*, 94, 19–35.
- Zellner, A. (1986), “On assessing prior distributions and Bayesian regression analysis with g -prior distributions,” in *Bayesian Inference and Decision Techniques: Essays in Honor of Bruno de Finetti*, Elsevier, pp. 233–243.
- Zellner, A. and Siow, A. (1980), “Posterior odds ratios for selected regression hypotheses,” in *Bayesian Statistics: Proceedings of the First International Meeting held in Valencia*, pp. 585–603.