Bayesian Ising Graphical Model for Variable Selection

Zaili FANG and Inyoung KIM

In this article, we propose a new Bayesian variable selection (BVS) approach via the graphical model and the Ising model, which we refer to as the “Bayesian Ising graphical model” (BIGM). The BIGM is developed by showing that the BVS problem based on the linear regression model can be considered as a complete graph and described by an Ising model with random interactions. There are several advantages of our BIGM: it is easy to (i) employ the single-site updating and cluster updating algorithm, both of which are suitable for problems with small sample sizes and a larger number of variables, (ii) extend this approach to nonparametric regression models, and (iii) incorporate graphical prior information. In our BIGM, the interactions are determined by the linear model coefficients, so we systematically study the performance of different scale normal mixture priors for the model coefficients by adopting the global-local shrinkage strategy. Our results indicate that the best prior for the model coefficients in terms of variable selection should place substantial weight on small, nonzero shrinkage. The methods are illustrated with simulated and real data. Supplementary materials for this article are available online.

Key Words: Cluster algorithm; Global-local shrinkage; Graphical model; Ising model; Mixture normals; Variable selection.

1. INTRODUCTION

Let us start from the standard multiple linear regression model \( y | \beta, \phi \sim N(X\beta, \phi^{-1}I) \), where \( y \) is an \( n \times 1 \) vector of the response variables, \( X = (x_1, \ldots, x_p) \) is an \( n \times p \) matrix of predictors, \( \beta = (\beta_1, \ldots, \beta_p)^T \) is a \( p \times 1 \) coefficient vector of the full model with \( \beta_j \) corresponding to the \( j \)th predictor, and \( \phi \) is the precision parameter. The inclusion or exclusion of the \( j \)th predictor in the model is represented by a binary indicator random variable \( \gamma_j \), where \( \gamma_j \in (0, 1) \). We denote the inclusion of predictor \( x_j \) with \( \gamma_j = 1 \), and otherwise we exclude it from the model. In recent years, incorporating prior network information of predictors into Bayesian variable selection (BVS) models has received substantial attention (Li and Zhang 2010; Tai, Pan, and Shen 2010; Monni and Li 2010; Stingo et al. 2011). In all these articles, the network information of the predictors are introduced through an
informative prior for $\gamma_j$’s, which is a binary random graph. However, none of these articles discuss treating variable selection as a graphical model with(out) a noninformative prior for $\gamma_j$’s. A binary random graphical model for the random vector $y = (\gamma_1, \ldots, \gamma_p)^T$ is represented by an undirected graph $G = (V, E)$, where $V$ represents the set of $p$ vertices or nodes corresponding to $p$ predictors and $E$ is a set of edges connecting neighboring nodes.

In this article, we base our approach on a reparameterized BVS model known as the KM model (Kuo and Mallick 1998). We develop the new BVS approach via the graphical model and the Ising model, which is referred to as “Bayesian Ising graphical model” (BIGM) for variable selection. We demonstrate that the linear regression model (KM model) is essentially a complete graphical model of $y$. A nice review about Ising model can be found in Newman and Barkema (1999) and Iba (2001).

Our contributions to this topic are in several aspects:

- First, by revealing that the binary Markov chain random process for $y$ on a graph can be modeled by the Ising model conditional on $\beta$ and $\phi$, we propose the BIGM. In a BIGM, the interactions between nodes are random and long range (each node is the neighbor of any other nodes). To have flexible interactions between nodes, we adopt the “shrink globally act locally” strategy (Polson and Scott 2011, 2012), which assigned scale normal mixture priors for the $\beta_j$’s (Barndorff-Nielsen, Kent, and Sørensen 1982; West 1987). By incorporating normal mixture priors of $\beta_j$ with long tail characteristic to the graph model of $\gamma_j$, the performance of BVS is significantly improved.

- Second, we develop a generalized cluster algorithm in which the cluster is formed with the random interactions among nodes. Possible approaches to explore the configuration space of $y$ in an Ising model are the cluster algorithm and a family of exchange Monte Carlo, parallel tempering, and simulated tempering algorithm (Iba 2001). However, the current cluster algorithms such as the Swendsen–Wang algorithm (Swendsen and Wang 1987) and Wolff algorithm (Wolff 1989) are constructed based on the graph prior for $y$ and only consider fixed interactions. Therefore, both are not applicable to the more general random complete graphical model. Furthermore, in our BIGM, it is straightforward to combine the graphical prior information of $y$, which helps to address the collinearity issues.

- Third, we extend our BIGM to the Bayesian sparse additive model (BSAM). There are only few articles discussing BVS under nonparametric regression (Smith and Kohn 1996; Reich, Storlie, and Bondell 2009; Scheipl 2011). We employ the Lancaster and Šalkauskas (LS) spline basis (Lancaster and Šalkauskas 1986; Chib and Greenberg 2010) to express the nonparametric function components such that we can simultaneously select an appropriate subset of the function components and estimate the flexible function curves. Additional advantages of BSAM include reducing the impact of the collinearity.

This article is organized as follows. In Section 2, we first introduce the KM hierarchical model and prior distributions of all model parameters at fixed shrinkage tuning parameter $b$, and then we discuss the connection between BVS and the binary random graphical
model. We finally express our model as the Ising model for $y$. In Section 3, we explain the single-site algorithm for sampling $y$, then present a cluster algorithm. In addition, we discuss how to incorporate a prior network information for $y$. Then in Section 4 and 5, we illustrate our model with simulations and real data analysis. Finally, in the last section, we provide concluding remarks and discuss other potential extensions of our model.

2. MODEL DESCRIPTION

2.1 BAYESIAN VARIABLE SELECTION WITH NORMAL MIXTURE PRIORS

We are interested in selecting a subset of predictors from the $p$ potential candidates by exploring the configuration space of $y = (y_1, \ldots, y_p)^T$. To implement the stochastic search for $\gamma_j$’s, in this article we consider the KM model, which is expressed as

$$\mathbf{y} = \mathbf{\beta_0}1 + \sum_{j=1}^{p} \gamma_j \mathbf{x}_j \mathbf{\beta}_j + \mathbf{\epsilon},$$

where $\mathbf{\epsilon} \sim \mathcal{N}(\mathbf{0}, \phi^{-1}I)$ is an iid noise vector. We standardize the dataset $X$ and center the response $\mathbf{y}$ such that $\sum_{i=1}^{n} x_{ij}^2 = 1$, $\sum_{i=1}^{n} x_{ij} = 0$, $j = 1, \ldots, p$ and $\sum_{i=1}^{n} y_i = 0$. Sampling procedure for $\mathbf{\beta}_0$ is straightforward with a normal prior, thus will not be discussed in details.

The prior of $\mathbf{\beta}$ plays an important role in our model. We seek a method that allows the $\beta_j$’s to be as flexible as possible to explore the configuration space so the variation of each $\beta_j$’s is modeled by an independent precision parameter. Meanwhile, we also seek to place a constrain on the overall variability of the interaction through a global shrinkage parameter, where for $b$. Therefore, we follow the “shrink globally act locally” scheme suggested by Polson and Scott (2011), which is easily implemented by imposing a hierarchical model: $[\beta_j | \tau_j, b] \sim \mathcal{N}(0, b^{-2} \tau_j^{-1})$, $[\tau_j] \sim p(\tau_j)$, $[\phi] \sim p(\phi)$, where $\tau_j$ is the precision parameter for the conditional normal prior of $\beta_j$ and plays the role of local tempering, and $b$ is the global shrinkage parameter to place a constrain on all $\tau_j$’s. $p(\tau_j)$ and $p(\phi)$ are the priors for $\tau_j$’s and $\phi$, respectively. Heaton and Scott (2010) also discussed similar hierarchical model under stochastic search variable selection (SSVS). With these settings, we can easily achieve the full conditional distribution for $\mathbf{\beta}$.

Prior selection for the $\tau_j$’s is critical too since it determines how the local behavior of the sampling process. There are several options for $p(\tau_j)$. In this article, we consider three widely known $p(\tau_j)$’s that result in three typical marginal $\beta_j$’s priors with heavy tails and/or heavy mass near zero. These marginal normal mixture priors of the $\beta_j$ are the Cauchy, Laplace, and horseshoe priors, which are achieved by assigning a Gamma prior $[\tau_j] \sim \mathcal{G}(1/2, 1/2)$, inverse gamma prior $[\tau_j] \sim \mathcal{IG}(1, 1/2)$, and half Cauchy prior $[\tau_j^{1/2}] \sim \mathcal{C}^+(0, 1)$ to $\tau_j$, respectively. Our theoretical analysis shows the horse shoe prior is the optimal choice (see the supplementary materials) because it maintains the heavy tail property for the density function of shrinkage parameter $\kappa_j = (\mathbf{\tau}_j \phi)/(1 + \mathbf{\tau}_j \phi)$ for a wide range of parameter $b$, thus the marginal selection probability $p(y_j = 1|b) = \pi_j^b/(1 + \pi_j^b)$ is more robust to $b$ than other two.
We simply assign a noninformative prior for \( \phi \) : \( \phi \sim \phi^{-1} \), so that the full conditional distribution of \( \phi \) becomes a gamma distribution. The details of full conditional distributions for updating \( \beta_j, \tau_j \), and \( \phi \) can be found in the supplementary materials.

2.2 BAYESIAN ISING GRAPHICAL MODEL

Given \( \beta \), consider the matrix of marginal regression functions \( R = (r_1, \ldots, r_p) = (\beta_1x_1, \ldots, \beta_px_p) \), with each column as the marginal regression vector for \( j \)th predictor vector, thus the full conditional distribution of \( \gamma \) with the prior \( \gamma \sim \exp(\sum_{i<j} W_{ij} \delta_{ij}) \) is \( p(\gamma | y, \beta, \phi) \propto \exp\left( \sum_{i<j} J^*_{ij} \delta_{ij} + \sum_j h^*_j \gamma_j \right) \), (2)

where the first summation is on all \( i < j, i, j = 1, \ldots, p \), \( \delta_{ij} = 1 \) if \( \gamma_i = \gamma_j \) otherwise \( \delta_{ij} = 0 \), and \( J^*_{ij} = J_{ij} + W_{ij} \). This is the Boltzmann distribution of the Ising model, \( \frac{1}{Z} \exp(-U(\gamma)) \), with \( U(\gamma) = -\sum_{i<j} J^*_{ij} \delta_{ij} - \sum_j h^*_j \gamma_j \), where \( Z = \sum_{\gamma} \exp(-U(\gamma)) \) is called the partition (normalized) function and \( U(\gamma) \) is called the “energy” of state \( \gamma \) given \( \beta \) and \( \phi \). \( J_{ij} = \phi \beta_i(x_i^T x_j) \beta_j \) is the element of the interaction matrix \( J = -\frac{\phi R^T R}{2} \), and \( h^*_j \) is the element of the vector \( h^* = \phi R^T (y - R1/2) \) named as the “external field.” Note that the naive uniform prior \( \gamma \sim (\frac{1}{2})^p \) is obtained for \( W_{ij} = 0 \) for all \( i, j \).

Model (2) is a typical Ising model well discussed in Iba (2001) except here it is a complete (with interaction among all nodes) graphic model with random external field (\( h^*_j \)). Figure S3 of the supplementary materials gives a diagram to understand the complete graphical model with more details.

3. ALGORITHM FOR UPDATING \( \gamma \)

3.1 SINGLE-SITE ALGORITHM

First, we consider a general Metropolis–Hastings (MH) one-step updating procedure. Denote the current state for \( \gamma_j \) as \( \gamma^0_j | \gamma_j \) and its flipped state \( \gamma^*_j | \gamma_j \), whether or not we move from \( \gamma^0_j | \gamma_j \) to \( \gamma^*_j | \gamma_j \) depends on the “energy” difference \( \Delta U = U(\gamma^*_j | \gamma_j) - U(\gamma^0_j | \gamma_j) \). \( \gamma_j \) denotes the vector of \( \gamma \) with \( \gamma_j \) excluded. We prefer the system in lower “energy” state since the lower the energy, the higher the probability. Thus if \( \Delta U \leq 0 \), the flipped state is accepted with probability 1. We treat the case \( \Delta U > 0 \) probabilistically, that is, with the probability of acceptance as \( p(\Delta U) = \exp(-\Delta U) \). These steps can be summarized by flipping the current state to its opposite with probability

\[
\min \{1, \exp(-\Delta U)\}.
\]

The detailed balance maintains and this MH updating is used in the Markov chain Monte Carlo (MCMC) Ising model sampling (Newman and Barkema 1999; Nott and Green 2004). In this article, unless otherwise specified, we adopt the one step MH updating (3) with other Gibbs samplers in all cases of single-site algorithm.
3.2 Cluster Algorithm

Beyond the single-site algorithm, the cluster algorithm is well established for simulating model (2) when $J_{ij}^*$'s and $h_j^*$'s are fixed (Swendsen and Wang 1987; Wolff 1989; Nott and Green 2004). In general, the cluster algorithm performs better than the single-site updating when $J_{ij}^*$ is fixed. However, as pointed out before, it is difficult to apply the cluster algorithm to the model (2) as there is a random external field $h_j^*$ and the coupling coefficients $J_{ij}^*$'s follow some unknown distribution with a nonnegligible dependence structure. Additionally, the nodes are connected with each other by so-called long-range interactions, and thus the system is a totally disordered complete graph. In this article, we propose a generalized single-cluster Monte Carlo algorithm, which is similar to Wolff’s clustering scheme but is also capable of handling long-range interactions and a random external field.

In the original SW and Wolff algorithm, clusters are formed through the bonding between paired nodes on a lattice with positive interactions. Unlike the usual Ising model on a one-dimension chain or two/three-dimension lattice, the complete graph model of the binary random process is fully connected. This indicates that the behavior of each node is determined according to the overall effects of all other nodes. The clustering dynamics must therefore incorporate this consideration.

We use $c$ to denote the cluster, and $\bar{c}$ as the complement of $c$. Within the cluster, there are two subclusters that are antialigned. We denote these two sub-clusters as $c_1$ and $c_0$ with $\gamma_{c_1} = 1$ and $\gamma_{c_0} = 0$, respectively. The cluster with only aligned nodes can be considered as the special case with zero nodes in one of the sub-cluster.

The question, then, is: given a particularly defined probability $p_a$ of adding a node to the cluster, what is the acceptance ratio that makes the flip of the cluster satisfy detailed balance. Also, how does one choose $p_a$ such that the average acceptance ratio is as large as possible? We derive following generalized Wolff algorithm based on these considerations.

1. Form the cluster.
   (a) Initialize the cluster set $c$ by randomly picking a seed node.
   (b) Examine the nodes in $\bar{c}$ one by one in a random order, add the node $j$ in $\bar{c}$ to the cluster with the probability

\[
p_{a,j} = \max \left\{ 1 - \exp \left[ \lambda (-1)^{\gamma_j} \left( \sum_{k \in c_1} J_{jk}^* - \sum_{l \in c_0} J_{jl}^* \right) \right], 0 \right\},
\]

and remove $j$ from $\bar{c}$ if $j$ added to $c$, where $0 \leq \lambda \leq 1$. Continue iteratively until no new sites added when each nodes in $\bar{c}$ has been examined.

2. Flip the nodes in cluster $c$ with probability

\[
\alpha (\mathbf{y}_c^0 \to \mathbf{y}_c^*) = \min \left\{ \exp \left[ (1 - \lambda) \sum_{j \in c} (-1)^{\gamma_j} \left( \sum_{k \in c_1} J_{jk}^* - \sum_{l \in c_0} J_{jl}^* \right) \right] + \sum_{j \in c_0} h_j^* - \sum_{j \in c_1} h_j^* , 1 \right\}.
\]
3. Flip the rest nodes in $\tilde{c}$ (if any are left) by the single-site updating method (3).

4. Update $\beta_j$’s, $\tau_j$’s, and $\phi$.

In Equations (4) and (5), $\lambda$ plays a role of partial clustering similar to Higdon (1998). When $\lambda = 1$, all interaction terms in (5) are annihilated, which means the cluster is completely decoupled from its neighbors. If $\lambda = 0$, then no clustering occurs, and the algorithm is reduced to the single-site algorithm. For the naive uniform prior, $W_{ij} = 0$, then $J^*_{ij}$ in (4) and (5) are reduced to $J_{ij}$.

The cluster algorithm including the process of forming and flipping the cluster can be better explained using the diagram of Figure S4 of the supplementary materials. The general idea is: whether or not a new node should be added is determined by the bonding between the new node and all nodes in the cluster, and to flip these nodes we have to cutoff the bonding of the cluster with all other nodes in the $\tilde{c}$ because in a complete graph the neighbors of a cluster are all the other nodes outside the cluster.

**Theorem 1.** With the probability of adding node to the cluster, $p_{a,j}$, and the probability of moving from the current configuration $\gamma^0_{\tilde{c}}$ to the flipped configuration $\gamma^*_{\tilde{c}}$, $\alpha(\gamma^0_{\tilde{c}} \rightarrow \gamma^*_{\tilde{c}})$, as defined as in the generalized Wolff algorithm, the algorithm is detail balanced and ergodic.

**Proof.** See A.1 of the supplementary materials.

It is easy to show that our algorithm is more general than Wolff’s in sense that it is applicable to complete graphs with random interaction. When applied to the Ising model on a lattice with positive fixed interaction $J$, where only interactions among the nearest neighbors count, our algorithm evaluates to the original Wolff algorithm: the cluster grows by adding bonds between nearest neighbors with probability $1 - \exp(-J)$.

As discussed in the above context, it is very straightforward to incorporate a graph prior for $\gamma$ in our BIGM, which is equivalent to placing the regularization on $\gamma$. Since two connected nodes with positive interaction tend to be selected or excluded together, only the prior graph for $\gamma$ with positive interaction ($W_{ij} \geq 0$) is meaningful. If we have the information that some selected nodes and their neighbors are all true nodes, then incorporating a graph prior with those nodes connected will improve the power to identify the nodes with small signal. Take, for example, a given true model, $y = \sum_{j \in S} x_j \beta_j$, where $S = 1, \ldots, k$ is sequential index up to $k$ and $k < p$. Obviously there are some information about the true variables such that there are $k$ sequential nodes that are true nodes, and $p - k$ sequential nodes that do not belong to the true model. Therefore, an Ising prior with a one-dimensional linear chain will be a very efficient prior since this prior reflects the information that sequential nodes are selected or excluded together.

**4. SIMULATION STUDY**

**4.1 CHOICE OF TUNING PARAMETER AND THE OPTIMAL NORMAL MIXTURE PRIOR**

A good choice of $b$ should separate the signals from the noise very well in terms of the marginal selection probability of $\gamma_j = 1$. Similar to the recommendations of Liang et al.
(2008), we can take some empirical choices, say, $b = \sqrt{n}$ or $b = \max(\sqrt{n}, p)$. Or, we can assign a prior to $b$, such as a uniform prior, so the final inference is robust to $b$. On the other hand, by choosing appropriate normal mixture prior for $\beta$, we are able to widen the range of $b$ where the signals are well separated from the noise. It turns out that the horse shoe prior is superior to the Cauchy prior, and the Cauchy is superior to the Laplace prior (see Section S1 and S3.1 in the supplementary materials). Thus, the rest of the article employs only the horse shoe prior unless otherwise stated for all simulations and analysis. In this section, we use simulations to demonstrate the dependence of selection probability on $b$ at difference $n$ and $p$.

The simulation is a simple linear regression model

$$y = \sum_{j \in S} \beta_j x_j + \epsilon,$$

where $x_j \sim N(0, 1)$, $j = 1, \ldots, p$ and $\epsilon \sim N(0, 1)$, and we consider the following settings:

**Model I**

A. $p = 1000, n = 200$, $\beta_j = 0.8$ if $j$ is odd; $\beta_j = 1.0$ if $j$ is even. $S = \{31, 91, \ldots, 931\} \cup \{60, 120, \ldots, 960\}$

B. $p = 1000, n = 500$, $\beta_j = 0.8$ if $j$ is odd; $\beta_j = 1.0$ if $j$ is even. $S = \{31, 91, \ldots, 931\} \cup \{60, 120, \ldots, 960\}$

**Model II**

A. $p = 500, n = 100$, $\beta_j = -0.8$ if $j$ is odd; $\beta_j = 0.8$ if $j$ is even. $S = \{31, 91, \ldots, 451\} \cup \{60, 120, \ldots, 480\}$

B. $p = 500, n = 500$, $\beta_j = -0.8$ if $j$ is odd; $\beta_j = 0.8$ if $j$ is even. $S = \{31, 91, \ldots, 451\} \cup \{60, 120, \ldots, 480\}$

Thus for Model I and Model II, the number of true $\beta_j$’s is the cardinality $|S| = 32$ and $|S| = 16$, respectively. Under each setting, we performed the single-site updates with a total of 8000 iterations and the first 3000 discarded as burn-in. The average $\gamma_j$’s over $N = 5000$ iterations is calculated as the marginal selection probabilities.

Figure 1(a) and 1(b) plots the marginal selection probability of all variables against $b$ for two settings of Model I. These visualizations give us a general idea where an optimal $b$ can be found. It turns out that $b$ can be either very large (representing small shrinkage) or very small (representing large shrinkage). For example, $b \approx 600$ in Region III or $b \approx 0.1$ in Region I (see Figure 1(c) and 1(d)). Region II is a moderate shrinkage area with $b$ between 1 and 10. However, if $n$ is much smaller than $p$ or the signals are weaker, then the variable selection becomes inconsistent for certain value of $b$ in Region II as shown in Figure 1(a). Thus, one suggestion of choosing $b$ is to avoid the moderate shrinkage area unless the collinearity or interaction among predictors is reduced.

Similar patterns are found in settings of Model II. However, because of that the signal signs are either positive or negative in these settings, the consistency conditions are changed. This results in that in the large shrinkage area (Region I), variable selection is no longer
consistent, while consistency maintains in the small shrinkage area (see Figure 2). This simulation shows that the choice of $b$ can be contrary to the usual choice in which small $b$ (large shrinkage) is preferred (Lin and Zhang 2006; Lykou and Ntzoufras 2012). This result is consistent to our choice of the horse shoe prior as the normal mixture prior of $\beta_j$’s, which has the long tail characteristic (corresponding to small shrinkage). Thus, another suggestion of choosing $b$ is to avoid small $b$, which leads to large shrinkage and inconsistency.

Figure 1. The profile curves of the selection probability of Model I A and B ((a) and (b)). Selection probability at two specific $b$ values for Model I A ((c) and (d)).

Figure 2. The profile curves of the selection probability of Model II A and B ((a) and (b)). Selection probability at two specific $b$ values for Model II A ((c) and (d)).
4.2 COMPARISON OF CLUSTER AND SINGLE-SITE ALGORITHM

In this section, we show that the performances of the cluster and single-site algorithms both are $b$-dependent through the simulations. The result indicates in certain region of $b$, one algorithm may outperform the other.

To demonstrate this, we consider the simple simulation with the same model as (6) with large signals $S = \{\beta_2, \beta_3, \beta_5, \beta_{10}\} = \{-4, 2, -1, 2.5\}$, $n = 200$ and we vary $p$ from 50 to 1500. We run the simulation with four representative $b$’s, two are large and two are small, so that we may compare the different behavior of the two algorithms across different shrinkage parameters.

To measure the mixing or correlation time, it is convenient to define the “magnetization,” $M^{(i)}$, which represents the average value of the binary random variable $\gamma_j$’s at $i$th sweep of the MCMC iteration.

$$M^{(i)} = \frac{1}{p} \sum_{j=1}^{p} \gamma_j^{(i)}. $$

Thus, the mixing time of the MCMC sampler can be measured using the time-delayed autocorrelation function (ACF) of the Monte Carlo chain of “magnetization,”

$$C(t) = \frac{\sum_{i=1}^{N-t} (M^{(i)} - \tilde{M})(M^{(i+t)} - \tilde{M})}{\sum_{i=1}^{N} (M^{(i)} - \tilde{M})^2}, $$

where $t$ is the lag or the iteration time from the origin, measured in Monte Carlo sweeps (MCS), and $\tilde{M}$ is the average magnetization over total $N$ iterations. One way to measure the mixing time is simply using the summation of the autocorrelation time, $\sum_{t=0}^{L} |C(t)|$, where $L$ is the maximum lag calculated.

For each $p$, we performed 15,000 iterations or sweeps for each setting and discarded the first 5000. From the remaining $N = 10,000$ sweeps, we calculated the autocorrelation function $C(t)$ up to $L = 100$ lags. Figure 3 shows the summation of absolute ACF time against the nodes size $p$ for $b = 0.03$, 0.17, 1141, and 2195.

From Figure 3(a), we can see the differences in behavior of the cluster and single-site algorithms. In the large shrinkage region, $b = 0.03$ or 0.17, the cluster algorithm has mixing time uniformly smaller than the single-site algorithm and the mixing time of the cluster algorithm is at least two times shorter. Note that as the node size increases, the mixing time for the entire algorithm first decreases slightly and then stabilizes. It may increases as $p$ goes larger still.

In the small shrinkage region where $b = 1141$ and 2195, as shown in Figure 3(b), we see different characteristics. First, the measured mixing time is much noisier than in Figure 3(a), but the trend against $p$ is clear. Second, unlike in the large shrinkage area, here we see that for both algorithms the mixing time increases as $p$ increases. Furthermore, when $p$ is small, the single-site algorithm has a shorter mixing time, which slows down very fast as $p$ increases. For example, when $p = 60$, the summation of autocorrelation function is only several MCS, but reaches almost 100 MCS when $p$ is large than 1500, which translates to an extremely slow MCMC process. On the other hand, although the cluster algorithm is
about two times slower when \( p \) is small, also slowing down with increasing \( p \), the mixing time increases at a smaller rate and reaches no more than 50 when \( p = 1500 \).

Hence, in general, we find that the cluster algorithm outperforms single-site algorithm in terms of mixing time. However, which algorithm should be used depends on the data. The single-site algorithm is much less time-consuming since the cluster algorithm spends time in forming the cluster. The overall computational time for the cluster algorithm is high when \( p > 1000 \). Additionally, in many situations, the mixing time may not be much worse for the single-site algorithm. Thus, we prefer using the single-site algorithm to achieve the results quickly.

4.3 COLLINEARITY AND COMPARISON WITH OTHER APPROACHES

In this section, we demonstrate how to handle the collinearity issue in Bayesian variable selection with two different approaches: reconstructing the design matrix and incorporating the prior information for \( \gamma \).

In the first approach, the marginal regression predictor \( x_j \) is replaced as \( r_j = f_j(x_j) = Z_j \beta_j \), where \( Z_j \) is a basis matrix for the \( j \)th predictor and \( \beta_j \) is a multivariate random vector. The model becomes \( y = \beta_0 1 + \sum_{j=1}^{p} \gamma_j f_j(x_j) + \epsilon \), which is also a BSAM. The BSAM does not only implement variable selection on the sparse additive functional components, but also reduce the collinearity among predictors.

In this article, we apply Lancaster and Šalkauskas (LS) basis for the natural cubic spline to construct \( Z_j \) (see the supplementary materials). With LS basis, we can consider the prior of \( \beta_j \) as a \( K_j - 1 \)-dimensional multivariate normal: \( [\beta_j | \tau_{ej}, \tau_{bj}] \sim N(0, b^2 T_j) \), where \( K_j \) is the knot size of the cubic spline and \( T_j \) is a diagonal matrix with diagonal elements as \((\tau_{ej}^{-1}, \tau_{bj}^{-1}, \ldots, \tau_{bj}^{-1}, \tau_{ej}^{-1})\). We treat the whole set of \( \{\tau_{ej}, \tau_{dj} : j = 1, \ldots, p\} \) independent parameters. Similarly, we can still assign \( G(1/2, 1/2) \), \( IG(1, 1/2) \), or \( C^+ (0, 1) \) as their priors. However, the marginal prior for \( \beta_j \) given \( b \) is no longer a simple Cauchy, Laplace, or horseshoe prior as the linear regression model, but will maintain similar properties.
The simulation example for the first approach is

\[ y = f_1(x_1) + f_2(x_2) + f_3(x_3) + f_4(x_4) + \epsilon, \]

where \( x_j = (w_j + tu)/(1 + t), \) \( j = 1, \ldots, p \) and \( w_1, \ldots, w_p \) and \( u \) are iid from Uniform \((0,1), and \( \epsilon \sim N(0, 1.74). \) Here \( \text{Corr}(x_i, x_j) = t^2/(1 + t^2) \) for \( i \neq j \) and we consider \( t = 0 \) and \( t = 1. \) \( t = 1 \) gives the correlation between two predictors around 0.5, representing a strong collinearity. This simulation is similar to Example 1 in Lin and Zhang (2006) but we use \( p = 10, 80, \) and 150. We also consider sample size of \( n = 100. \) Functions \( f_j \)'s have following forms:

\[
\begin{align*}
    f_1(x) &= x, \\
    f_2(x) &= (2x - 1)^2, \\
    f_3(x) &= \sin(2\pi x)/[2 - \sin(2\pi x)], \\
    f_4(x) &= 0.1 \sin(2\pi x) + 0.2 \cos(2\pi x) + 0.3 \sin^2(2\pi x) + 0.4 \cos^3(2\pi x) \\
    &\quad + 0.5 \sin^3(2\pi x).
\end{align*}
\]

In this simulation, we employ the independent \( G(1/2, 1/2) \) prior for each \( \tau_{ej} \) and \( \tau_{dj} \) only. We fix \( K_j = 7, j = 1, \ldots, p \) in this simulation. In total, we use 6000 iterations with the single-site algorithm, discarding the first 2000 for all settings.

In Figure S6 of the supplementary materials, we plot the selection probability profile curves for \( t = 0 \) and \( t = 1 \) for the simulation setting \( p = 80. \) We also plot the estimated function components of four true functions and demonstrate how close they are to the true functions in Figure S7 and S8 of the supplementary materials. More discussion is shown in Section 3.2 of the supplementary materials.

To examine the performance in variable selection and the estimation accuracy, 500 simulation runs have been employed for \( p = 10, 80, \) and 150, respectively. We calculated seven statistics: “False Positive Rate (FP-rate),” “False Negative Rate (FN-rate),” “Model Size (MS),” and “Squared Error (SE)” of the four true functions, where FP-rate = \( \frac{\#\text{False Positive}}{\#\text{False Positive} + \#\text{True Negative}}, \) FN-rate = \( \frac{\#\text{False Negative}}{\#\text{False Negative} + \#\text{True Positive}}, \) and SE = \( \sum_i^n (f_{j,i} - \hat{f}_{j,i})^2 / n. \)

The estimated function is calculated by \( \hat{f}_j = Z_j E(\beta_j | y_j = 1), j \in \text{true nodes}. \) Since it can happen that \( p(y_j = 1|y) = 0 \) for any true function components, we simply estimate \( f_j \) by \( \hat{f}_j = 0 \) for the four true nodes if \( p(y_j = 1|y) = 0 \) in each run. The SE Statistics can be used to assess the accuracy of the estimation of the nonlinear function \( f_j \) because the smaller the SE, the closer the estimation \( \hat{f}_j \) is to the true function \( f_j. \) The average and standard deviation of those statistics over 500 runs are reported in Table 1 and compared with the component selection and smoothing operator (COSSO) (Lin and Zhang 2006).

As shown in Table 1, the results for our method is rather robust to \( p. \) For each \( t, \) all statistics are similar across different \( p \)'s with the exception of a slight increase in the mean and standard deviation of those statistics. On the other hand, we see that COSSO only performs well for small \( p. \) When \( p = 80 \) (we get no result for \( p = 150 \) using COSSO R package), all of the statistics increase significantly. The SEs are especially large for the four true function components, which means that COSSO cannot estimate those function components correctly. Our method is also pretty robust to the collinearity too: the performances are similar for \( t = 0 \) and \( t = 1, \) except for the slight increase in the values of FN-rate and
Table 1. Simulation results of sparse additive model (7) for 500 runs

|        | t | p   | FP-rate   | FN-rate   | MS  | f₁SE       | f₂SE       | f₃SE       | f₄SE       |
|--------|---|-----|-----------|-----------|-----|------------|------------|------------|------------|
| BIGM   | 0 | 10  | 0.00(0.02)| 0.00(0.03)| 3.99(0.17) | 0.07(0.05) | 0.16(0.06) | 0.18(0.08) | 0.74(0.27) |
|        | 80| 0.00(0.01)| 0.00(0.03)| 4.16(0.48) | 0.07(0.05) | 0.15(0.06) | 0.18(0.08) | 0.70(0.27) |
|        | 150| 0.00(0.02)| 0.01(0.04)| 4.81(5.52) | 0.08(0.07) | 0.16(0.08) | 0.17(0.11) | 0.73(0.44) |
|        | 1 | 10  | 0.00(0.01)| 0.09(0.10)| 3.56(0.54) | 0.08(0.07) | 0.18(0.09) | 0.18(0.10) | 0.79(0.40) |
|        | 80| 0.00(0.01)| 0.09(0.11)| 3.68(0.66) | 0.09(0.08) | 0.18(0.08) | 0.16(0.07) | 0.77(0.39) |
|        | 150| 0.01(0.03)| 0.11(0.11)| 4.48(7.17) | 0.10(0.10) | 0.18(0.09) | 0.18(0.10) | 0.80(0.40) |
| COSSO  | 0 | 10  | 0.00(0.01)| 0.00(0.00)| 4.00(0.06) | 0.07(0.04) | 0.05(0.04) | 0.11(0.06) | 0.32(0.13) |
|        | 80| 0.07(0.08)| 0.18(0.07)| 9.85(11.2) | 0.17(0.28) | 0.79(0.11) | 1.55(0.32) | 5.28(0.58) |
|        | 1 | 10  | 0.01(0.03)| 0.04(0.08)| 3.86(0.48) | 0.07(0.07) | 0.26(0.10) | 0.14(0.10) | 2.00(1.00) |
|        | 80| 0.10(0.09)| 0.19(0.10)| 12.5(14.0) | 0.36(0.40) | 0.37(0.16) | 1.05(0.44) | 4.67(0.54) |

SEs. In general, we see that our method works very well for BSAM even for the cases of large $p$ and large collinearity in both variable selection and function component estimation.

The second approach to reduce the collinearity problem is to incorporate prior information for $\gamma_j$ introduced in Section 3.2. To demonstrate this, we consider the simulation similar to model (6) with $p=100, n=100, \beta_j = 0.4$ if $j$ is odd, $\beta_j = 0.8$ if $j$ is even, and $S = \{1, 2, \ldots, 15\}$. The collinearity is introduced by generating $x_j = (w_j + tu)/(1+t)$ with $w_j, u \sim N(0, 1)$, and $t = 1$.

This example is special in the sense that its true predictor set $S$ and false signal set $\bar{S}$ both are continuous in their node index. Obviously, the simplest prior network information is a linear chain: any node’s two neighbors are most likely to align with this node. While this is not true for neighbored nodes 15 and 16, the discontinuity has a small effect on the system as a whole. Keeping this in mind, we consider the linear chain prior for the nodes.

We also compare the results to the same simulation of two other variable selection approaches: SSVS and g-prior. To have a fair comparison, three Bayesian models are constructed with similar hierarchical structures (see the supplementary material). The performance of all models with/without $\gamma$ prior information is measured by area under receiver operating characteristic (ROC) (AUC). AUC (0 ∼ 1) represents the power to classify each variable as signal or noise given a cutoff of selection probability (AUC = 1 means perfect separation of signal and noise). Figure 4(b) gives AUC against $b$ curves. First, our BIGM has the widest range of $b$ where the model is able to separate the signals and noise perfectly (AUC = 1), while g-prior model has the narrowest range. Second, for each model, incorporating the linear chain prior of $\gamma$ does improve the performance: at most of $b$ the one with $\gamma$ prior has higher AUC. The results also support our choice of KM hierarchical model (instead of SSVS) and normal mixture prior (instead of g-prior) for BIGM, and demonstrate how prior information of $\gamma$ can significantly improve the overall performance, particularly when strong multi-collinearity exists.
5. REAL DATA ANALYSIS

We introduce two real data analysis with our approach: The Ozone Data and Type II diabetes data. For Ozone data, we apply the BVS on additive model. Because of context limitation, the analysis results for Ozone data can be found in the supplementary materials.

5.1 GENE SELECTION IN PATHWAY DATA

Mootha et al. (2003) presented a pathway-based analysis to test a priori defined pathways for association with the diabetic disease. A pathway is a predefined set of genes that serve a particular cellular or physiological function. A genetic pathway can be expressed by a graph to represent the gene network within this pathway. Mootha et al. (2003) identified several significant pathways including “oxidative phosphorylation” and “alanine-and-aspartate metabolism.” However, even with those significant pathways identified, gene selection in microarray data analysis is still difficult because the effects of alterations in gene expression are modest due to the large number of genes, small sample sizes, and variability between subjects.

The data contain gene expressions from \( n = 35 \) subjects, 17 normal and 18 Type II diabetes patients. We merged three pathways of interest, “oxidative phosphorylation,” “alanine-and-aspartate metabolism,” and “glutamate-metabolism” into one graph with a total of \( p = 173 \) nodes. Some nodes are different probe sets of the same gene, so the gene names are identical. The graph with 173 nodes is a subgraph of the corresponding merged graph obtained from the KEGG database. The continuous response \( y \) is the glucose level.

The top left plot of Figure 5 shows the network of our merged gene set. Note that the prior required for our graph model is undirected graph with only positive interactions. We see that most of the nodes are independent in this dataset, and that there are only three genetic clusters. Because of this, if we apply the cluster algorithm and use the adjacency matrix \( \Lambda \) based on the network information into expression (4), we will end up with a few
nodes in the same genetic cluster that could potentially form the clusters for the algorithm. Therefore, we consider the following interaction matrix \( W = \{w_{ij}\lambda_{ij}\} \) for the prior of \( \gamma \) with adjacency matrix \( \Lambda = \{\lambda_{ij}\} \) as

\[
\lambda_{ij} = 1, \quad i, j = 1, \ldots, p
\]

\[
w_{ij} = \begin{cases} 
  w, & i \notin S \text{ or } j \notin S \\
  w + \Delta w, & i \in S \text{ and } j \in S,
\end{cases}
\]  

(9)

where \( S \) represents one of the three genetically networked gene clusters in the pathway network, \( w \) and \( \Delta w \) are small positive numbers stand for the strength of the interaction in the prior and the difference of two types of interaction. If \( \Delta w = 0 \), we can consider (9) as a baseline graph prior for \( \gamma \), which is a complete graph with positive fixed interaction. Since we also vary \( b \) to have an overall view of the selection probability, it is necessary to have \( w \to 0 \) when \( b \to 0 \) since with large shrinkage, \( J_{ij} \to 0 \) and we do not want \( w_{ij} \) to dominate the graph interaction. One convenient way to avoid this to express \( w \) as
$w = w_0 \Phi[\log(b)]$, which approaches 0 as $b \to 0$ and reaches the maximum $w_0$ for large $b$, where $\Phi(\cdot)$ is the CDF of standard normal. Note that the choice of $w$ is involved in the consideration of the so-called phase transition (Li and Zhang 2010). If $w$ is too large, all the nodes will always be connected, which leads to either selection of all nodes or none of the nodes.

Now we consider $\Delta w \neq 0$, say $\Delta w = 5w$, so that we incorporate the genetic network information into the graph prior. $\Delta w$ cannot be too large, otherwise those genes in the genetic clusters will always be aligned, which means that they will all have a small selection probability. We therefore choose $w_0 \approx 0.01$ as small as possible to avoid the phase transition phenomenon, but $w_0$ must be large enough to reduce the interaction between signals and noise caused by small sample size. With this selection and for large $b$, we have the prior interaction $w_{ij} \approx 0.01$ for two nodes not in the same genetic cluster, and $w_{ij} \approx 0.06$ for two nodes in the same cluster.

In Figure 5(a) and 5(b), we plot the selection probability profile curves with (non)informative priors for $\gamma$. The improvement of incorporating prior information for the graph model is obvious. We ran the cluster algorithm for total $N = 40,000$ iterations, and discarded the first 10,000 as burn-in. Therefore, the selection probability is calculated by taking the mean of $\gamma$ over 30,000 iterations. In Figure 5(a), with a noninformative prior for $\gamma$, we can see that all the curves are mixed for the moderate value of $b$. On the other hand, in Figure 5(b), with an informative prior for $\gamma$ defined as (9), the profile curves are much “cleaner” even at moderate $b$. Around $b = 8.5$ we see a bunch of curves is clearly distinguishable from the rest.

We then fixed $b = 8.5$ and ran the cluster algorithm for $N = 60,000$ iterations with the first 20,000 discarded. With this shrinkage parameter, the prior interaction parameter was $w_{ij} \approx 0.06$ for $i$ and $j$ in the genetic cluster and $w_{ij} \approx 0.01$ otherwise. The selection probability for all nodes are shown in the top right of Figure 5, where we take a cutoff probability as 0.2 and identify six nodes that have relative high selection probabilities. Among those nodes, UQCRB has the largest selection probability for the entire range of $b$, so it is easy to identify UQCRB as the most significant gene. We also select other five genes, COX8, ATP5G2 (two probe sets), ATP5H, and CRAT at $b = 8.5$. All the genes selected except CRAT are from “oxidative phosphorylation” pathway, which is related to ATP synthesis. It is well known that ATP plays an importance role in Type II diabetic disease. CRAT is from “alanine-and-aspartate metabolism” pathway.

6. DISCUSSION

The goal of this article is to present the BIGM from two major viewpoints. The first is how to sample the “in” or “out” binary random variable. We pointed out that BVS can be considered as a binary random process on a complete graph given noninformative prior for $\gamma_j$’s, and we compared the single-site and generalized Wolff cluster updating algorithm. The other viewpoint deals with how to construct the interaction matrix of the complete graph, which is implemented by sampling the linear model coefficient $\beta_j$’s through the scale mixtures of normal priors.
We also discussed the marginal selection probability profile under different shrinkage parameter and compared three prior settings for $\beta_j$, which represent three typical situations of shrinkage proportion. Our BIMG method possesses the advantages of simplicity, easy implementation, and straightforward extension. For example, the BIMG is very easy to extend to Bayesian sparse model by representing the nonparametric function components $f_j$ as a linear combination of the basis matrix $f_j = Z_j \beta_j$. We can then employ the group selection of vector $\beta_j$. Another example of an extension of our method is the incorporation of network information for $\gamma$. Although this article does not focus on how to construct the prior network structure information, the simulation and real data analyses show that it is easy to incorporate the prior graph information and improve the performance of the BIMG.

However, this article only starts with a different view about BVS, and further research could include but is not limited to the following questions. (i) We simply choose $b$ by an empirical method, further theoretical study of choosing $b$ and the appearance of the Lindley’s paradox are preferred. (ii) Fixing $b$ limits the performance of our method, we can adopt a remedy similar to exchange Monte Carlo by running parallel MCMCs at two or more $b$’s with some $b$’s small and others large, and exchanging their configuration according to a certain probability that remains detailed balanced. (iii) It is difficult to construct a meaningful network for the prior of $\gamma$. This keeps an open question discussed by Li and Zhang (2010) and Monni and Li (2010).

SUPPLEMENTARY MATERIALS

The supplementary materials provide information including: theoretical analysis and dynamic properties of the marginal selection odds $p(\gamma_j = 1)/p(\gamma_j = 0)$; full conditional distributions to update the parameters of the Bayesian computing for BIMG, SSVS, and g-prior; additional simulation results about comparison of Cauchy, the horseshoe and Laplace priors, and Baysian sparse additive model; finally, additional real data analysis on ozone data.

**REFERENCES**

Barndorff-Nielsen, O., Kent, J., and Sørensen, M. (1982), “Normal Variance-Mean Mixtures and z Distribution,” *International Statistical Review*, 50, 145–159. [590]

Chib, S., and Greenberg, E. (2010), “Additive Cubic Spline Regression With Dirichlet Process Mixture Errors,” *Journal of Econometrics*, 156, 322–336. [590]

Heaton, M., and Scott, J. (2010), “Bayesian Computation and the Linear Model,” in *Frontiers of Statistical Decision Making and Bayesian Analysis*, eds. M. H. Chen, D. K. Dey, P. Mueller, D. Sun, and K. Ye, New York: Springer, pp. 527–545. [591]

Higdon, D.M. (1998), “Auxiliary Variable Methods for Markov Chain Monte Carlo With Applications,” *Journal of the American Statistical Association*, 93, 585–595. [594]

Iba, Y. (2001), “Extended Ensemble Monte Carlo,” *International Journal of Modern Physics C*, 12, 623–656. [590,592]
Kuo, L., and Mallick, B. (1998), “Variable Selection for Regression Models,” *Sankhyā: The Indian Journal of Statistics B*, 60, 65–81. [590]

Lancaster, P., and Šalkauskas, K. (1986), *Curve and Surface Fitting: An Introduction*, San Diego, CA: Academic Press. [590]

Li, F., and Zhang, N.R. (2010), “Bayesian Variable Selection in Structured High-Dimensional Covariates Spaces with Applications in Genomics,” *Journal of the American Statistical Association*, 105, 1202–1214. [589,603,604]

Liang, F., Paulo, R., Molina, G., Clyde, M.A., and Berger, J.O. (2008), “Mixtures of g Priors for Bayesian Variable Selection,” *Journal of the American Statistical Association*, 103, 410–423. [595]

Lin, Y., and Zhang, H.H. (2006), “Component Selection and Smoothing in Multivariate Nonparametric Regression,” *The Annals of Statistics*, 34, 2272–2297. [596,599]

Lykou, A., and Ntzoufras, I. (2012), “On Bayesian Lasso Variable Selection and the Specification of the Shrinkage Parameter,” *Statistics and Computing*, 23, 361–390. DOI: 10.1007/s11222-012-9316-x. [596]

Monni, S., and Li, H. (2010), “Bayesian Methods for Network-Structured Genomics Data,” in *Frontiers of Statistical Decision Making and Bayesian Analysis*, eds. M. H. Chen, D. K. Dey, P. Mueller, D. Sun, and K. Ye, New York: Springer, pp. 303–326. [589,604]

Mootha, V. K., Lindgren, C. M., Eriksson, K., Subramanian, A., Sihag, S., Lehar, J., Puigserver, P.,Carlsson, E., Riddlerstrale, M., Laurila, E., Houstis, N., Daly, M. J., Patterson, N., Mesirov, J. P., Golub, T. R., Tamayo, P., Spiegelman, B., Lander, E. S., Hirschhorn, J. N., Altshuler, D., and Groop, L. C. (2003), “PGC-l Alpha-Responsive Genes Involved in Oxidative Phosphorylation are Cooperately Downregulated in Human Diabetes,” *Nature Genetics*, 34, 267–273. [601]

Newman, M. E. J., and Barkema, G. T. (1999), *Monte Carlo Methods in Statistical Physics*, New York: Oxford University Press. [590,592]

Nott, D. J., and Green, P. J. (2004), “Bayesian Variable Selection and the Swenden-Wang Algorithm,” *Journal of Computational and Graphical Statistics*, 13, 141–157. [592]

Polson, N. G., and Scott, J. G. (2011), “Shrink Globally, Act Locally: Sparse Bayesian Regularization and Prediction,” in *Proceedings of the 9th Valencia World Meeting on Bayesian Statistics*, Oxford University Press, pp. 501–538. [590,591]

——— (2012), “Local Shrinkage Rules, Lévy Processes and Regularization Regression,” *Journal of the Royal Statistical Society*, Series B, 74, 287–311. [590]

Reich, B. J., Storlie, C. B., and Bondell, H. D. (2009), “Variable Selection in Bayesian Smoothing Spline ANOVA Models: Application to Deterministic Computer Codes,” *Journal of Econometrics*, 51, 110–119. [590]

Scheipl, F. (2011), “spikeSlabGAM: Bayesian Variable Selection, Model Choice and Regularization for Generalized Additive Mixed Models in R,” *Journal of Statistical Software*, 43, 1–24. [590]

Smith, M., and Kohn, R. (1996), “Nonparametric Regression Using Bayesian Variable Selection,” *Journal of Econometrics*, 75, 317–343. [590]

Stingo, F. C., Chen, Y. A., Tadesse, M. G., and Vannucci, M. (2011), “Incorporating Biological Information into Linear Models: A Bayesian Approach to the Selection of Pathways and Genes,” *The Annals of Applied Statistics*, 5, 1978–2002. [589]

Swendsen, R. H., and Wang, J. S. (1987), “Nonuniversal Critical Dynamics in Monte Carlo Simulations,” *Physical Review Letters*, 58, 86–88. [590,593]

Tai, F., Pan, W., and Shen, X. (2010), “Bayesian Variable Selection in Regression with Networked Predictors,” in *High-Dimensional Data Analysis*, eds. T. Cai and X. Shen, Singapore: World Scientific, pp. 147–165. [589]

West, M. (1987), “On Scale Mixtures of Normal Distributions,” *Biometrika*, 74, 646–648. [590]

Wolff, U. (1989), “Collective Monte Carlo Updating for Spin Systems,” *Physical Review Letters*, 62, 361–364. [590,593]