

Evolutionary Stochastic Search for Bayesian Model Exploration

Leonard Bottolo*, and Sylvia Richardson†

Abstract. Implementing Bayesian variable selection for linear Gaussian regression models for analysing high dimensional data sets is of current interest in many fields. In order to make such analysis operational, we propose a new sampling algorithm based upon Evolutionary Monte Carlo and designed to work under the “large p , small n ” paradigm, thus making fully Bayesian multivariate analysis feasible, for example, in genetics/genomics experiments. Two real data examples in genomics are presented, demonstrating the performance of the algorithm in a space of up to 10,000 covariates. Finally the methodology is compared with a recently proposed search algorithms in an extensive simulation study.

Keywords: Evolutionary Monte Carlo, Fast Scan Metropolis-Hastings scheme, linear Gaussian regression models, variable selection

1 Introduction

This paper is a contribution to the methodology of Bayesian variable selection for linear Gaussian regression models, an important problem which has been much discussed both from a theoretical and a practical perspective (see [Chipman et al. 2001](#); [Clyde and George 2004](#); [O’Hara and Sillanpää 2009](#) for literature reviews). Recent advances have been made in two directions, unravelling the theoretical properties of different choices of prior structure for the regression coefficients ([Fernández et al. 2001](#); [Liang et al. 2008](#)) and proposing algorithms that can explore the huge model space consisting of all the possible subsets when there are a large number of covariates, using either MCMC or other search algorithms ([Kohn et al. 2001](#); [Dellaportas et al. 2002](#); [Hans et al. 2007](#)).

In this paper, we propose a new sampling algorithm for implementing the variable selection model, based on tailoring ideas from Evolutionary Monte Carlo ([Liang and Wong 2000](#); [Jasra et al. 2007](#); [Wilson et al. 2009](#)) in order to overcome the known difficulties that MCMC samplers face in a high dimension multimodal model space: enumerating the model space becomes rapidly unfeasible even for a moderate number of covariates. For a Bayesian approach to be operational, it needs to be accompanied by an algorithm that samples the indicators of the selected subsets of covariates, together with any other parameters that have not been integrated out. Our new algorithm for searching through the model space has many generic features that are of interest *per se* and can be easily coupled with any prior formulation for the variance-covariance of the

*MRC Clinical Sciences Centre and Centre for Biostatistics, Imperial College London, London, UK <mailto:l.bottolo@imperial.ac.uk>

†Centre for Biostatistics, Imperial College London, London, UK <mailto:sylvia.richardson@imperial.ac.uk>

regression coefficients. We illustrate this by implementing g -priors for the regression coefficients as well as independent priors: in both cases the formulation we adopt is general and allows the specification of a further level of hierarchy on the priors for the regression coefficients, if so desired.

The paper is structured as follows. In Section 2, we present the background of Bayesian variable selection, reviewing briefly alternative prior specifications for the regression coefficients, namely g -priors and independent priors. Section 3 is devoted to the description of our MCMC sampler which uses a wide portfolio of moves, including two proposed new ones. Section 4 demonstrates the good performance of our new MCMC algorithm in a variety of real and simulated examples with different structures on the predictors. In Section 4.2 we complement the results of the simulation study by comparing our algorithm with the recent Shotgun Stochastic Search algorithm of Hans et al. (2007). Finally Section 5 contains some concluding remarks.

2 Background

2.1 Variable selection

Let $y = (y_1, \dots, y_n)^T$ be a sequence of n observed responses and $x_i = (x_{i1}, \dots, x_{ip})^T$ a vector of predictors for y_i , $i = 1, \dots, n$, of dimension $p \times 1$. Moreover let X be the $n \times p$ design matrix with i th row x_i^T . A Gaussian linear model can be described by the equation

$$y = \alpha \mathbf{1}_n + X\beta + \varepsilon,$$

where α is an unknown constant, $\mathbf{1}_n$ is a column vector of ones, $\beta = (\beta_1, \dots, \beta_p)^T$ is a $p \times 1$ vector of unknown parameters and $\varepsilon \sim N(0, \sigma^2 I_n)$.

Suppose one wants to model the relationship between y and a subset of x_1, \dots, x_p , but there is uncertainty about which subset to use. Following the usual convention of only considering models that have the intercept α , this problem, known as variable selection or subset selection, is particularly interesting when p is large and parsimonious models containing only a few predictors are sought to gain interpretability. From a Bayesian perspective the problem is tackled by placing a constant prior density on α and a prior on β which depends on a latent binary vector $\gamma = (\gamma_1, \dots, \gamma_p)^T$, where $\gamma_j = 1$ if $\beta_j \neq 0$ and $\gamma_j = 0$ if $\beta_j = 0$, $j = 1, \dots, p$. The overall number of possible models, defined through γ , grows exponentially with p and selecting the best model that predicts y is equivalent to find one over the 2^p subsets that form the model space.

Given the latent variable γ , a Gaussian linear model can therefore be written as

$$y = \alpha \mathbf{1}_n + X_\gamma \beta_\gamma + \varepsilon, \tag{1}$$

where β_γ is the non-zero vector of coefficients extracted from β , X_γ is the design matrix of dimension $n \times p_\gamma$, $p_\gamma \equiv \gamma^T \mathbf{1}_p$, with columns corresponding to $\gamma_j = 1$. We will assume that, apart from the intercept α , x_1, \dots, x_p contains no variables that would be included in every possible model and that the columns of the design matrix have all been centred with mean 0.

It is recommended to treat the intercept separately and assign it a constant prior: $p(\alpha) \propto 1$, [Fernández et al. \(2001\)](#). When coupled with the latent variable γ , the conjugate prior structure of (β_γ, σ^2) follows a normal-inverse-gamma distribution

$$p(\beta_\gamma | \gamma, \sigma^2) = N(m_\gamma, \sigma^2 \Sigma_\gamma) \tag{2}$$

$$p(\sigma^2 | \gamma) = p(\sigma^2) = InvGa(a_\sigma, b_\sigma) \tag{3}$$

with $a_\sigma, b_\sigma > 0$. Some guidelines on how to fix the value of the hyperparameters a_σ and b_σ are provided in [Kohn et al. \(2001\)](#). Taking into account (1), (2), (3) and the prior specification for α , the joint distribution of all the variables (based on further conditional independence conditions) can be written as

$$p(y, \gamma, \alpha, \beta_\gamma, \sigma^2) = p(y | \gamma, \alpha, \beta_\gamma, \sigma^2) p(\alpha) p(\beta_\gamma | \gamma, \sigma^2) p(\sigma^2) p(\gamma). \tag{4}$$

The main advantage of the conjugate structure (2) and (3) is the analytical tractability of the marginal likelihood whatever the specification of the prior covariance matrix Σ_γ :

$$\int p(y | \gamma, \alpha, \beta_\gamma, \sigma^2) p(\alpha) p(\beta_\gamma | \gamma, \sigma^2) p(\sigma^2) d\alpha d\beta_\gamma d\sigma^2 \propto |X_\gamma^T X_\gamma + \Sigma_\gamma^{-1}|^{-1/2} |\Sigma_\gamma|^{-1/2} (2b_\sigma + S(\gamma))^{-(2a_\sigma + n - 1)/2}, \tag{5}$$

where $S(\gamma) = C - M^T K_\gamma^{-1} M$, $C = (y - \bar{y}_n)^T (y - \bar{y}_n) + m_\gamma^T \Sigma_\gamma^{-1} m_\gamma$, $M = X_\gamma^T (y - \bar{y}_n) + \Sigma_\gamma^{-1} m_\gamma$, $K_\gamma = X_\gamma^T X_\gamma + \Sigma_\gamma^{-1}$ ([Brown et al. 1998](#)) with $\bar{y}_n = 1/n \sum_{i=1}^n y_i$.

While the mean of the prior (2) is usually set equal to zero, $m_\gamma = 0$, a neutral choice ([Chipman et al. 2001](#); [Clyde and George 2004](#)), the specification of the prior covariance Σ_γ matrix leads to at least two different classes of priors:

- When $\Sigma_\gamma = gV_\gamma$, where g is a scalar and $V_\gamma = (X_\gamma^T X_\gamma)^{-1}$, it replicates the covariance structure of the likelihood giving rise to so called g -priors first proposed by [Zellner \(1986\)](#).
- When $\Sigma_\gamma = cV_\gamma$, but $V_\gamma = I_{p_\gamma}$ the components of β_γ are conditionally independent and the posterior covariance matrix is driven towards the independence case.

We will adopt the notation $\Sigma_\gamma = \tau V_\gamma$ as we want to cover both prior specification in a unified manner. Thus in the g -prior case, $\Sigma_\gamma = \tau (X_\gamma^T X_\gamma)^{-1}$ while in the independent case, $\Sigma_\gamma = \tau I_{p_\gamma}$. We will refer to τ as the *variable selection coefficient* for reasons that will become clear in the next section.

To complete the prior specification in (4), $p(\gamma)$ must be defined. A complete discussion about alternative priors on the model space can be found in [Chipman \(1996\)](#) and

Chipman et al. (2001). Here we adopt the beta-binomial prior illustrated in Kohn et al. (2001)

$$p(\gamma) = \int p(\gamma|\omega) p(\omega) d\omega = \frac{B(p_\gamma + a_\omega, p - p_\gamma + b_\omega)}{B(a_\omega, b_\omega)} \quad (6)$$

with $p_\gamma \equiv \gamma^T \mathbf{1}_p$, where the choice $p(\gamma|\omega) = \omega^{p_\gamma} (1-\omega)^{p-p_\gamma}$ implicitly induces a binomial prior distribution over the model size and $p(\omega) = \omega^{a_\omega-1} (1-\omega)^{b_\omega-1} / B(a_\omega, b_\omega)$. The hypercoefficients a_ω and b_ω can be chosen once $E(p_\gamma)$ and $V(p_\gamma)$ have been elicited.

2.2 Priors for the variable selection coefficient τ

g-priors

It is a known fact that *g*-priors have two attractive properties. Firstly they possess an automatic scaling feature (Chipman et al. 2001; Kohn et al. 2001). The second feature that makes *g*-priors particularly appealing is the simple structure of the marginal likelihood (5) with respect to the constant τ which becomes

$$\propto (1+\tau)^{-p_\gamma/2} (2b_\sigma + S(\gamma))^{-(2a_\sigma+n-1)/2}, \quad (7)$$

where, if $m_\gamma = 0$, $S(\gamma) = (y - \bar{y}_n)^T (y - \bar{y}_n) - \frac{\tau}{1+\tau} (y - \bar{y}_n)^T X_\gamma (X_\gamma^T X_\gamma)^{-1} X_\gamma^T (y - \bar{y}_n)$. Despite the simplicity of (7), the choice of the constant τ for *g*-priors is complex, see Fernández et al. (2001), Cui and George (2008) and Liang et al. (2008).

Historically, the first attempt to build a comprehensive Bayesian analysis where the degree of shrinkage adapts to different scenarios dates back to Zellner and Siow (1980) who proposed to place a prior distribution on τ . Zellner-Siow priors, Z-S hereafter, can be thought as a mixture of *g*-priors and an inverse-gamma prior on τ , $\tau \sim \text{InvGa}(1/2, n/2)$, leading to

$$p(\beta_\gamma | \gamma, \sigma^2) \propto \int N(0, \sigma^2 \tau (X_\gamma^T X_\gamma)^{-1}) p(\tau) d\tau. \quad (8)$$

For alternative priors, see also Cui and George (2008), Liang et al. (2008) and Maruyama and George (2008).

Independent priors

When all the variables are defined on the same scale, independent priors represent an attractive alternative to *g*-priors. The likelihood marginalised over α , β_γ and σ^2 becomes

$$p(y|\gamma) \propto \tau^{-p_\gamma/2} |X_\gamma^T X_\gamma + \tau I_{p_\gamma}|^{-1/2} (2b_\sigma + S(\gamma))^{-(2a_\sigma+n-1)/2}. \quad (9)$$

If $m_\gamma = 0$, $S(\gamma) = (y - \bar{y}_n)^T (y - \bar{y}_n) - (y - \bar{y}_n)^T X_\gamma (X_\gamma^T X_\gamma + \tau I_{p_\gamma})^{-1} X_\gamma^T (y - \bar{y}_n)$. Note that (9) is computationally more demanding than (7) due to the extra determinant operator.

It is common practice to standardise the predictor variables, taking $\tau = 1$ in order to place appropriate prior mass on reasonable values of the regression coefficients (Hans et al. 2007). Another approach, illustrated in Bae and Mallick (2004), places a prior distribution on τ_j without standardising the predictors.

3 MCMC sampler

In this section we propose a new sampling algorithm that overcomes the known difficulties faced by MCMC schemes when attempting to sample a high dimension multimodal space. We discuss in a unified manner the general case where a hyperprior on the variable selection coefficient τ is specified. This encompasses the g -prior and independent prior structure as well as the case of fixed τ if a point mass prior is used.

The multimodality of the model space is a known issue in variable selection and several ways to tackle this problem have been proposed in the past few years. Liang and Wong (2000) suggest an extension of parallel tempering called Evolutionary Monte Carlo, EMC hereafter, Nott and Green (2004) introduce a sampling scheme inspired by the Swendsen-Wang algorithm while Jasra et al. (2007) extend EMC methods to varying dimension algorithms. Finally Hans et al. (2007) propose, when $p > n$, a new stochastic search algorithm, Shotgun Stochastic Search, SSS hereafter, to explore models that are in the same neighbourhood in order to quickly find the best combination of predictors.

We propose to solve the issue related to the multimodality of model space (and the dependence between γ and τ) along the lines of EMC, applying some suitable parallel tempering strategies directly on $p(y, \gamma | \tau)$. The basic idea of parallel tempering, PT hereafter, is to weaken the dependence of a function from its parameters by adding an extra one called “temperature”. Multiple Markov chains, called “population” of chains, are run in parallel, where a different temperature is attached to each chain, their state is tentatively swapped at every sweep by a probabilistic mechanism and the latent binary vector γ of the non-heated chain is recorded. The different temperatures have the effect of flattening the likelihood. This ensures that the posterior distribution is not trapped in any local mode and that the algorithm mixes efficiently, since every chain constantly tries to transmit information about its state to the others. EMC extends this idea, encompassing the positive features of PT and genetic algorithms inside a MCMC scheme. In the following, we first define the expanded state-space of EMC, which is common with PT, and then we describe the specific MCMC moves that characterise EMC.

Since β and σ^2 are integrated out, only two parameters need to be sampled from the joint distribution $p(\tau) \prod_{l=1}^L [p(y | \gamma_l, \tau) p(\gamma_l)]^{1/t_l}$, namely the latent binary vector and the variable selection coefficient. In this set-up the full conditionals to be considered are

$$[p(\gamma_l | \dots)]^{1/t_l} \propto [p(y | \gamma_l, \tau)]^{1/t_l} [p(\gamma_l)]^{1/t_l} \quad (10)$$

$$p(\tau | \dots) \propto p(\tau) \prod_{l=1}^L [p(y | \gamma_l, \tau)]^{1/t_l}, \quad (11)$$

where L is the number of chains in the population and t_l , $1 = t_1 < t_2 < \dots < t_L$, is the temperature attached to the l th chain while the population γ corresponds to a set of chains that are retained simultaneously.

At each sweep of our algorithm, first the population γ in (10) is updated using a variety of moves inspired by genetic algorithms: “local moves”, the ordinary Metropolis-Hastings or Gibbs update on every chain, and “global moves” that include: i) crossover operator, i.e. partial swap of the current state between different chains (selected on the base of some probabilistic measures of distance between them); ii) exchange operator, full state swap between chains. Both local and global moves are important although global moves are crucial because they allow the algorithm to jump from one local mode to another. At the end of the update of γ , τ is then sampled using (11). Conditions for convergence of EMC algorithms are well understood and illustrated for instance in Jasra et al. (2007).

The implementation of EMC that we propose in this paper includes several novel aspects: the use of a wide range of moves including two new ones, a local move, based on the Fast Scan Metropolis-Hastings sampler, particularly suitable when p is large and a bold global move that exploits the pattern of correlation of the predictors. Moreover, we developed an efficient scheme for tuning the temperature placement that capitalises the effective interchange between the chains. Another new feature is to use a Metropolis-within-Gibbs with adaptive proposal for updating τ , as the full conditional (11) is not available in closed form.

3.1 EMC sampler for γ

In what follows, we will only sketch the rationale behind all the moves that we found useful to implement and discuss further the benefits of the new specific moves in Section 4.1. For the “large p , small n ” paradigm and complex predictor spaces, we believe that using a wide portfolio of moves is needed and offers better guarantee of mixing.

From a notational point of view, we will use the double indexing $\gamma_{l,j}$, $l = 1, \dots, L$ and $j = 1, \dots, p$ to denote the j th latent binary indicator in the l th chain. Moreover we indicate by $\gamma_l = (\gamma_{l,1}, \dots, \gamma_{l,p})^T$ the vector of binary indicators that characterises the state of the l th chain of the population $\gamma = (\gamma_1, \dots, \gamma_L)$.

Local moves and Fast Scan Metropolis Hastings sampler

Given τ , we first tried the simple MC³ idea of Madigan and York (1995), also used by Brown et al. (1998) where add/delete and swap moves are used to update the latent binary vector γ_l . For an add/delete move, one of the p variables is selected at random and if the latent binary value is 0 the proposed new value is 1 or *vice versa*. However, when $p \gg p_{\gamma_l}$, where p_{γ_l} is the size of the current model for the l th chain, the number of sweeps required to select by chance a binary indicator with a value of 1 follows a geometric distribution with probability p_{γ_l}/p which is much smaller than $1 - p_{\gamma_l}/p$ to select a binary indicator with a value of 0. Hence, the algorithm spends most of the time

trying to add rather than delete a variable. Note that this problem also affects RJ-type algorithms (Dellaportas et al. 2002). On the other hand, Gibbs sampling (George and McCulloch 1993) is not affected by this issue since the state of the l th chain is updated by sampling from

$$[p(\gamma_{l,j} = 1 | y, \gamma_{l,\setminus j}, \tau)]^{1/t_l} \propto \exp \left\{ \left(\log p(y | \gamma_{l,j}^{(1)}, \tau) + \log p(\gamma_{l,j} = 1 | \gamma_{l,\setminus j}) \right) / t_l \right\}, \tag{12}$$

where $\gamma_{l,\setminus j}$ indicates for the l th chain all the variables, but the j th, $j = 1, \dots, p$ and $\gamma_{l,j}^{(1)} = (\gamma_{l,1}, \dots, \gamma_{l,j-1}, \gamma_{l,j} = 1, \gamma_{l,j+1}, \dots, \gamma_{l,p})^T$. The main problem related to Gibbs sampling is the large number of models it evaluates if a full Gibbs cycle or any permutation of the indices is implemented at each sweep. Each model requires the direct evaluation, or at least the update, of the time consuming quantity $S(\gamma)$, equation (7) or (9), making practically impossible to rely solely on the Gibbs sampler when p is very large. However, as sharply noticed by Kohn et al. (2001), it is wasteful to evaluate all the p updates in a cycle because if p_{γ_l} is much smaller than p and given $\gamma_{l,j} = 0$, it is likely that the sampled value of $\gamma_{l,j}$ is again 0.

When p is large, we thus consider instead of the standard MC³ add/delete, swap moves, a novel Fast Scan Metropolis-Hastings scheme, FSMH hereafter, specialised for EMC/PT. The idea behind the FSMH move is to use an additional acceptance/rejection step (which is very fast to evaluate) to choose the number of indices where to perform the Gibbs-like step: the novelty of our FSMH sampler is that the additional probability used in the acceptance/rejection step is based not only on the current chain model size p_{γ_l} , but also on the temperature t_l attached to the l th chain. To find an alternative scheme to a full Gibbs sampler and to save computational time, our strategy is to evaluate the time consuming marginal likelihood (5) in no more than approximately $\left\lceil \tilde{\theta}_{l,\bullet}^{(1)}(1/t_l)(p - p_{\gamma}) + \tilde{\theta}_{l,\bullet}^{(0)}(1/t_l)p_{\gamma} \right\rceil$ times per cycle in the l th chain (assuming convergence is reached), where $\tilde{\theta}_{l,\bullet}^{(1)}(1/t_l)$ is the averaged probability (over the set $\{j : j \in \gamma_{l,j} = 0\}$) to select a variable to be added in the acceptance/rejection step which is function of the temperature t_l and similarly for $\tilde{\theta}_{l,\bullet}^{(0)}(1/t_l)$, the averaged probability (over the set $\{j : j \in \gamma_{l,j} = 1\}$) of selecting a variable to be deleted ($\lceil \cdot \rceil$ indicates the integer part). Since for chains attached to lower temperatures $\tilde{\theta}_{l,\bullet}^{(0)}(1/t_l) \gg \tilde{\theta}_{l,\bullet}^{(1)}(1/t_l)$, the algorithm proposes to update *almost all* binary indicators with value 1, while it selects at random a group of approximately $\left\lceil \tilde{\theta}_{l,\bullet}^{(1)}(1/t_l)(p - p_{\gamma}) \right\rceil$ binary indicators with value 0 to be updated. At higher temperatures since $\tilde{\theta}_{l,\bullet}^{(0)}$ and $\tilde{\theta}_{l,\bullet}^{(1)}$ become more similar, the number of models evaluated in a cycle increases because much more binary indicators with value 0 are updated. It turns out that our FSMH sampler is computationally less demanding than a full Gibbs sampling on all $\gamma_{l,j}$ and does not suffer from the problem highlighted before for MC³ and RJ-type algorithms when $p \gg p_{\gamma_l}$. Full details of the FSMH scheme are given in the Appendix A.1.1, while evaluation of it and comparison with MC³ embedded in EMC are presented in Section 4.1.

Global moves

Global moves are conceived to allow the algorithm to escape from local modes. To do so they need to reach the right compromise between the boldness of the moves (distance of the chains) and the efficiency (acceptance rate of the moves). Hence all the global moves start with a carefully designed selection of the chains where the global operator is applied.

Crossover operator

The crossover operator is aimed at swapping partial state of two different chains. The first step of crossover move (selection step) consists of selecting the pair of chains (l, r) to be operated on. We use renormalised Boltzmann weights to increase the chance that the two selected chains will give rise, after the crossover, to a new configuration of the population with higher posterior probability. To be precise, we compute a probability equal to the weight of the “Boltzmann probability”, $p_t(\gamma_l | \tau) = \exp\{f(\gamma_l | \tau)/t\} / F_t$, where $f(\gamma_l | \tau) = \log p(y | \gamma_l, \tau) + \log p(\gamma_l)$ is the log transformation of the full conditional (10) assuming $t_l = 1 \forall l$, $l = 1, \dots, L$, and $F_t = \sum_{l=1}^L \exp\{f(\gamma_l | \tau)/t\}$ for some specific temperature t , and then rank all the chains according to this. Among the population of chains, we identify a group \tilde{L} , $1 \leq \tilde{L} \leq L$, of chains with high Boltzmann weights. We then overweight this subpopulation by assigning a positive but small probability to the chains that do not belong to this group, and renormalise all the probabilities. Finally, using these probabilities, we select at random without replacement from *all* chains, the two chains where the crossover operator will be performed.

Suppose that two new latent binary vectors are then generated from the selected chains according to some crossover operators described in the next paragraph. The new proposed population of chains $\gamma' = (\gamma_1, \dots, \gamma'_l, \dots, \gamma'_r, \dots, \gamma_L)$ is accepted with probability

$$\alpha(\gamma \rightarrow \gamma') = \min \left\{ 1, \frac{\exp\{f(\gamma'_l | \tau)/t_l + f(\gamma'_r | \tau)/t_r\} Q_t(\gamma' \rightarrow \gamma | \tau)}{\exp\{f(\gamma_l | \tau)/t_l + f(\gamma_r | \tau)/t_r\} Q_t(\gamma \rightarrow \gamma' | \tau)} \right\}, \quad (13)$$

where $Q_t(\gamma \rightarrow \gamma' | \tau)$ is the proposal density which is defined as the product of the selection probability and the crossover operator probability (see [Liang and Wong \(2000\)](#)).

In the following we will assume that four different crossover operators are selected at random at every EMC sweep: 1-point crossover, uniform crossover, adaptive crossover ([Liang and Wong 2000](#)) and a novel block crossover. Of these four moves, the uniform crossover which “shuffles” the binary indicators along all the selected chains is expected to have a low acceptance, but to be able to genuinely traverse regions of low posterior probability. The block crossover essentially tries to swap a group of variables that are highly correlated and can be seen as a multi-points crossover whose crossover points are not random but defined from the correlation structure of the covariates. In practice the block crossover is defined as follows: one variable is selected at random with probability $1/p$, then the pairwise correlation $\rho(X_j, X_{j'})$ between the j th selected predictor and each of the remaining covariates, $j' = 1, \dots, p$, $j' \neq j$, is calculated. We then retain for the block crossover all the covariates with positive (negative) pairwise correlation with

X_j such that $|\rho(X_j, X_{j'})| \geq \rho_0$. The threshold ρ_0 is chosen with consideration to the specific problem, but we fixed it at 0.25. Evaluation of block crossover and comparisons with other crossover operators are presented on a real data example in Section 4.1.

Exchange operator

The exchange operator can be seen as an extreme case of crossover operator, where the first proposed chain receives the whole second chain state $\gamma'_l = \gamma_r$, and *vice versa*. In order to achieve a good acceptance rate, the exchange operator is usually applied on adjacent chains in the temperature ladder, which limits its capacity for mixing. To obtain better mixing, we implemented two different variations: the first one is based on Jasra et al. (2007) and the related idea of delayed rejection (Green and Mira 2001); the second, a bolder “all-exchange” move, is based on a precalculation of all the $L(L-1)/2$ exchange acceptance rates between all chains pairs (Calvo 2005). Full relevant details are presented in Appendix A.1.1. Both of these bold moves perform well in the real data applications (see Section 4.1) and simulated examples (see Section 4.2) thus contributing to the efficiency of the algorithm.

Temperature placement

As noted by Goswami and Liu (2007), the placement of the temperature ladder is the most important ingredient in population based MCMC methods. We propose a procedure for the temperature placement which has the advantage of simplicity while preserving good accuracy. First of all, we fix the size L of the population. In doing this, we are guided by several considerations: the complexity of the problem, i.e. $E(p_\gamma)$, the size of the data and computational limits. We have experimented and we recommend to fix $L \geq 3$. Even though some of the simulated examples had $p_\gamma \simeq 20$ (Section 4.2), we found that $L = 5$ was sufficient to obtain good results. In our real data examples (Section 4.1), we used $L = 4$ guided by some prior knowledge on $E(p_\gamma)$. Secondly, we fix at an initial stage, a temperature ladder according to a geometric scale such that $t_{l+1}/t_l = b$, $b > 1$, $l = 1, \dots, L$ with b relatively large, for instance $b = 4$. To subsequently tune the temperature ladder, we then adopt a strategy based on monitoring the acceptance rate of the delayed rejection exchange operator towards a target of 0.5 (Liu 2001; Jasra et al. 2007) during a fixed burn-in period. After the burn-in the temperature ladder stays fixed. Details of the implementation are reported in Appendix A.1.1

3.2 Adaptive Metropolis-within-Gibbs for τ

Various strategies can be used to avoid having to sample from the posterior distribution of the variable selection coefficient τ . The easiest way is to integrate it out through a Laplace approximation (Tierney and Kadane 1986) or using a numerical integration such as quadrature on an infinite interval. We do not pursue these strategies and the reasons can be summarised as follows. If the Laplace approximation is performed in order to integrate out τ , it is required that every chain has its own value of the variable selection coefficient τ_l . In this set-up, *equilibrium* in the product space is difficult to

reach because the posterior distribution of γ_l depends, through the marginal likelihood obtained using the Laplace approximation, on the *chain specific value* of the posterior mode for $\tau_l, \hat{\tau}_{\gamma_l}$. Since the strength of X_{γ_l} to predict the response is weakened for chains attached to high temperatures, it turns out that for these chains, $\hat{\tau}_{\gamma_l}$ is likely to be close to zero. When the variable selection coefficient is very small, the marginal likelihood dependence on X_{γ_l} decreases even further, see for instance (7), and chains attached to high temperatures will experience a very unstable behaviour, making the convergence in the product space hard to reach.

In this paper the convergence is reached in the product space $p(\tau) \prod_{l=1}^L [p(\gamma_l | y, \tau)]^{1/t_l}$, i.e. the whole population is conditioned on a value of τ *common to all chains*. This strategy will alleviate the problems highlighted before allowing for faster convergence and better mixing among the chains. The procedure just described comes with an extra cost, i.e. sampling the value of τ . However, this step is inexpensive in relation to the cost required to sample $\gamma_l, l = 1, \dots, L$. There are several strategies that can be used to sample τ from (11). We found useful to apply the idea of adaptive Metropolis-within-Gibbs described in Roberts and Rosenthal (2009). Conditions for the asymptotic convergence and ergodicity are guaranteed as we enforce the *diminishing adaptive condition*, i.e. the transition kernel stabilises as the number of sweeps goes to infinity and the *bounded convergence condition*, i.e. the convergence time of the kernel is bounded in probability. In our set-up using an adaptive proposal to sample τ has several benefits; amongst others it avoids the known problems faced by the Gibbs sampler when the prior is proper, but relatively flat (Natarajan and McCulloch 1998) as can happen for Z-S priors when n is large or for the independent case considered by Bae and Mallick (2004). Moreover, given an upper limit on the number of sweeps, the adaptation guarantees a better exploration of the tails of $p(\tau | y)$ than with a fixed proposal. For details of the implementation and discussion of conditions for convergence, see Appendix A.1.2.

3.3 ESS algorithm

In the following, we refer to our proposed algorithm, Evolutionary Stochastic Search as ESS. If g -priors are chosen the algorithm is denoted as ESS g , while we use ESS i if independent priors are selected (the same notation is used when τ is fixed or given a prior distribution). Without loss of generality, we assume that the response vector and the design matrix have both been centred and, in the case of independent priors, given the lack of any autoscaling feature, we also assume that the design matrix is rescaled. Based on the two full conditionals (10) and (11) and the local and global moves introduced earlier, our ESS algorithm can be summarised as follows.

- Given τ , sample the population's states γ from the two steps:
 - (i) With probability 0.5 perform local move and with probability 0.5 apply at random one of the four crossover operators: 1-point, uniform, block and adaptive crossover. If local move is selected, use FSMH sampling scheme independently for each chain. Moreover every 100 sweeps apply on the first chain a complete scan by a Gibbs sampler.

- (ii) Perform the delayed rejection exchange operator or the all-exchange operator with equal probability. During the burn-in, only select the delayed rejection exchange operator.
- When τ is not fixed but has a prior $p(\tau)$, given the latent configuration $\gamma = (\gamma_1, \dots, \gamma_L)$, sample τ from an adaptive Metropolis-within-Gibbs sampling (Section 3.2).

From a computational point of view, we used the same fast form for updating $S(\gamma)$ as Brown et al. (1998), based on the QR-decomposition. Besides its numerical benefits, QR-decomposition can deal with the case $p_\gamma \geq n$. This avoids having to restrict the search to models with $p_\gamma < n$, and helps mixing during the burn-in phase.

4 Performance of ESS

4.1 Real data examples

Regulation of gene expression

The first real data example is an application of linear regression to investigate genetic regulation. To discover the genetic causes of variation in the expression (i.e. transcription) of genes, gene expression data are treated as a quantitative phenotype while genotype data (SNPs) are used as predictors, a type of analysis known as expression Quantitative Trait Loci (eQTL). In this context it is important to distinguish *cis*-eQTLs, where the genetic control points (SNPs) are located close to the location of the transcribed gene, from *trans*-acting eQTLs which lie on a different chromosome. The latter are a more complex form of regulation which is of major interest, but rather hard to detect. Besides SNPs that are located close to the transcript, we only expect a few additional control points associated with the variation of gene expression for a particular transcript (Hübner et al. 2005). Given this prior biological knowledge on the genetic regulation, in this example we assume $E(p_\gamma) = 4$ and $V(p_\gamma) = 2$.

Here we focus on the ability of ESS to find a parsimonious set of predictors in an animal data set (Hübner et al. 2005), where the number of observations, $n = 29$, is small with respect to the number of covariates $p = 1,421$. This situation, where $n \ll p$, is quite common in animal experiments since environmental sources of variation are controlled as well as the biological diversity of the sample. For illustration, we report the analysis of one gene expression response, DEXH58, a negative regulator of host innate immune defense against viruses located in chromosome 17. We apply ESS g with and without the hyperprior on τ , see Table 1–eQTL. In the former case, thanks to the adaptive proposal, the Markov chain for τ mixes very well reaching an overall acceptance rate which is close to the target value 0.44 as discussed in Roberts and Rosenthal (2009). Convergence is not a problem since the trace of the proposal’s standard deviation stabilises quickly and well inside the bounded conditions.

In both cases a good mixing among the $L = 4$ chains is obtained (Figure 1, top panels, ESS g with $\tau = 29$). Although in the case depicted in Figure 1 with fixed τ , the convergence is reached in the product space $\prod_{l=1}^L [p(\gamma_l | y)]^{1/t_l}$, by visual inspection

we see that each chain *marginally* reaches its *equilibrium* with respect to the others; moreover, thanks to the automatic tuning of the temperature placement during the burn-in, the distributions of the chains log posterior probabilities (assuming $t_l = 1 \forall l, l = 1, \dots, L$) overlap nicely, allowing effective exchange of information between the chains. Table 1–eQTL, confirms that the automatic temperature selection works well (with and without the hyperprior on τ) reaching an acceptance rate for the monitored exchange (delayed rejection) operator close to the selected target of 0.50 which is in agreement with the recommendation of Iba (2001). The all-exchange operator shows a higher acceptance rate, while, in contrast to Jasra et al. (2007), the overall crossover acceptance rate is reasonably high: in our experience the good performance of the crossover operator is both related to the selection step (Section 3.1) and the new block crossover which shows an acceptance rate far higher than the others. Finally the computational time on the same desktop computer (see details in Appendix A.2.3) is rather similar with or without the hyperprior τ , 28 and 30 minutes respectively for 25,000 sweeps with 5,000 as burn-in.

Looking at the output of ESS with variable τ , the best model visited has two eQTLs, but there seems to be support also for three eQTLs (see Figure 2 (c)). Interestingly the best model visited is a bigenic model with two *trans*-eQTLs in chromosome 15 which are within 5 cM apart. The same area on chromosome 15 is also known to control IRF7 gene which is the central hub of an inflammatory gene network. The presence of additional visited polygenic models that point besides chromosome 15 locus, to loci on chromosome 8 and 20 suggests a potential interaction of DEXH58 gene with other genes involved in the innate immune inflammatory response.

The main difference among the two implementations of ESS $_g$ with and without a hyperprior on τ is related to the posterior model size: when τ is fixed at $\tau = 29$ (Unit Information Prior, Fernández et al. 2001), there is more uncertainty and support for larger models, see Figure 2 (a). The best model visited (and the corresponding $R_\gamma^2 = 1 - S(\gamma)/y^T y$) is the same for both version of ESS $_g$, but, when a hyperprior on τ is implemented, the “stability index” which indicates how much the algorithm persists on the first chain top 1,000 (not unique) visited models ranked by the posterior probability (Appendix A.2.3), shows a higher stability, see Table 1–eQTL. In this case, having a data-driven level of shrinkage helps the search algorithm to better discriminate among competing models.

Metabolomics Quantitative Trait Loci

Our second example is related to the application of model (1) in another genomics example: 10,000 SNPs are linked to metabolomics data (Dumas et al. 2007), an example of a so-called mQTL experiment. The predictors come from a genome-wide analysis of a candidate gene study for ALAT enzyme elevation in liver (Kindmark et al. 2008) with the Mass Spectrography metabolomics data derived from the same sample (Andersson et al. 2009). A suitable dimension reduction of the spectral data is performed to divide the spectra in regions or bins and \log_{10} -transformation is applied in order to normalise the signal.

We present the key findings on the performance of ESS with regards to a particular

		Mode($p_\gamma y$)	$E(\tau y)$	R_γ^{2*}	$\overline{R_\gamma^{2**}}$	Stability
eQTL	ESSg, $\tau = 29$	2	–	0.716	0.704	0.257
	ESSg with $p(\tau)$	2	20.576	0.716	0.689	0.099
mQTL	ESSg, $\tau = 50$	2	–	0.843	0.843	≈ 0
	ESSg with $p(\tau)$	2	63.577	0.843	0.843	≈ 0
		Crossover	DR Exch.	ALL Exch.	Acc. rate τ	Time (min.)
eQTL	ESSg, $\tau = 29$	0.214	0.534	0.671	–	28
	ESSg with $p(\tau)$	0.243	0.585	0.711	0.438	30
mQTL	ESSg, $\tau = 50$	0.214	0.514	0.669	–	302
	ESSg with $p(\tau)$	0.226	0.571	0.717	0.434	309

Table 1: Performance of ESSg with and without the hyperprior on τ for the first real data example, eQTL analysis, and second example, mQTL analysis. R_γ^{2*} and $\overline{R_\gamma^{2**}}$ correspond to “ R_γ^{2*} : $\max p(\gamma | y)$ ” and “ $\overline{R_\gamma^{2**}}$: 1,000 largest $p(\gamma | y)$ ” respectively. The former indicates the coefficient of determination for the (first chain) best model visited according to the posterior probability $p(\gamma | y)$, while the latter shows the average R_γ^{2*} for the (first chain) top 1,000 (not unique) visited models ranked by the posterior probability. “Stability” is defined as the standard deviation of R_γ^{2*} for the (first chain) top 1,000 (not unique) visited models (smaller values indicate better performance of the algorithm). In the bottom part of the Table, acceptance rate for specific moves are given. “DR Exch.” and “ALL Exch.” stands for “delayed rejection exchange” and “all-exchange” move respectively.

metabolite bin that discriminates between the disease status of the clinical trial’s participants, but the same comments can be extended to the analysis of the whole data set, where we regressed every metabolites bin *versus* the genotype data ($n = 50$ and $p = 10,000$). In this very challenging “large p , small n ” case, we still found an efficient mixing of the chains (see Table 1–mQTL). Note that for this case the posterior mean of τ , 63.6, is a little larger than the Unit Information Prior, $\tau = n$, although the influence of the hyperprior is less important than in the previous real data example, see Figure 2 (b).

As expected in view of the very large number of predictors, in the mQTL example the computational time is quite large, around 5 hours for 20,000 sweeps after a burn-in of 5,000, but as shown in Table 1 by the “stability index” (≈ 0), we believe that the number of iterations chosen exceeds what is required in order to visit faithfully the model space. For such large data analysis tasks, parallelisation of the code could provide big gains of computer time and would be ideally suited to our multiple chains approach.

When analysing the output of ESSg (assuming *a priori* $E(p_\gamma) = 5$ and $V(p_\gamma) = 3$), it emerges that two SNPs are alternatively associated with a set of other SNPs in a bigenic or polygenic manner (see Figure 2 (d)), suggesting that these SNPs could act as “pivot” markers in the regulation of the metabolite variation. These two SNPs are located inside

COBLL1 gene, chromosome 2, and CORIN gene, chromosome 4, respectively, with the latter indirectly associated in genetic literature with various effects on liver function.

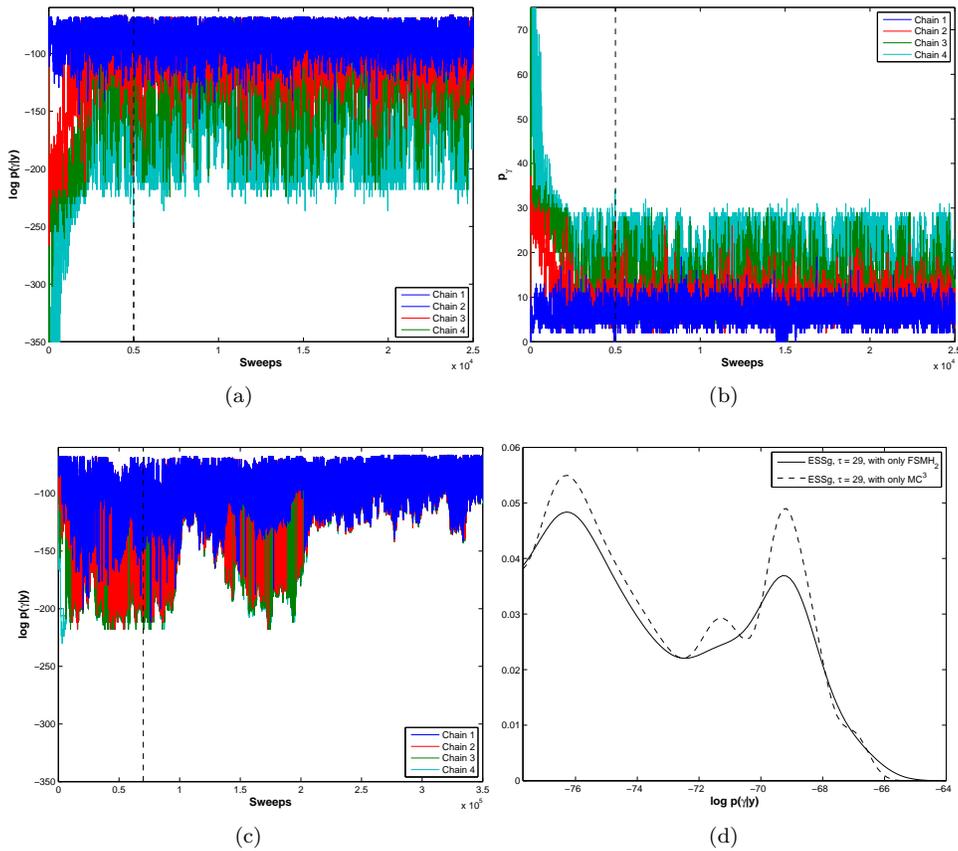


Figure 1: Top panels: (a) trace plot of the log posterior probability, $\log p(\gamma|y)$, and (b) model size, p_γ , across sweeps for the first real data example, eQTL analysis, using ESSg with $\tau = 29$ and FSMH as local move. Vertical dashed lines indicate the end of the burn-in. Bottom panels: (c) trace plot of the log posterior probability when MC^3 is used as a local move; (d) kernel densities of $\log p(\gamma|y)$ for the retained chain in the 25 replicates of the analysis when only FSMH and only MC^3 are used as a local move respectively. Plot restricted to regions of high posterior probability.

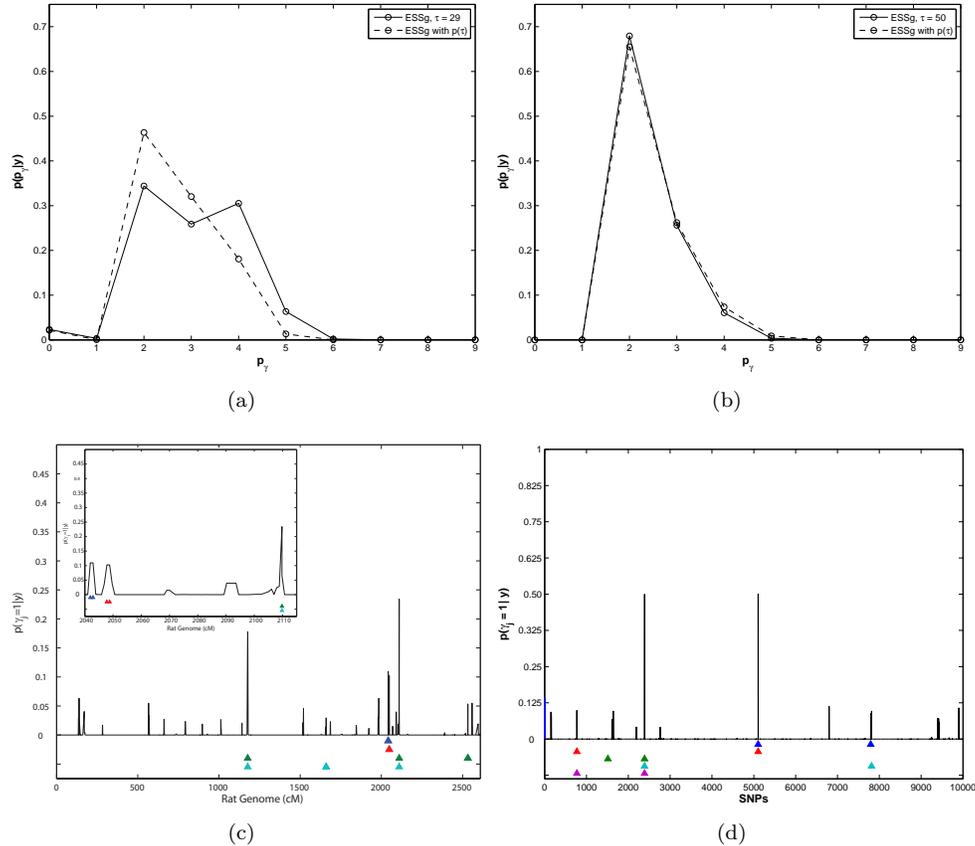


Figure 2: (a) Posterior model size for the first real data example, eQTL analysis: black solid line for $ESSg$ with τ fixed at 29 and black dashed line for $ESSg$ with Z-S prior. (b) Posterior model size for mQTL analysis, second real data example, using $ESSg$ with fixed and random τ . (c) Back lines: marginal posterior probability of inclusion for each SNP derived by $ESSg$ with a hyperprior on g for the first real data example, eQTL analysis. From top to bottom: triangles (with different colours) indicate SNPs in the first (blue), second (red), third (green) and fourth (cyan) best visited model with the fifth model equals to the null model (no triangles). Insertion magnifies a region spanning 100 cM (2040 – 2140) which helps to distinguish the genetic control points for the first and second best visited model, where the blue triangles pinpoint two closely located markers at chromosome 15 and similarly for the red triangles. (d) Back lines: marginal posterior probability of inclusion for each SNP derived by $ESSg$ with a hyperprior on g for the second real data example, mQTL analysis. From top to bottom triangles (with different colours) indicate SNPs in the first (blue), second (red), third (green), fourth (cyan) and fifth (magenta) best visited model.

Benefits of FSMH, block crossover and hyperprior on τ

In the regulation of gene expression example, we also evaluate the superiority of our ESS algorithm, and in particular the FSMH scheme and the block crossover, with respect to more traditional EMC implementations illustrated for instance in [Liang and Wong \(2000\)](#). Albeit we believe that using a wide portfolio of different moves enables any searching algorithm to better explore complicated model spaces, we compared: (i) ESS g with only FSMH as local move vs ESS with only MC³ as local move; (ii) ESS g with only block crossover vs ESS g with only 1-point, only uniform and only adaptive crossover respectively; (iii) ESS g with and without a hyperprior on τ . To avoid dependency of the results on the initialisation of the algorithm, we replicated the analysis 25 times. Moreover, to make the comparison fair, in experiment (i) we run the two versions of ESS g for a different number of sweeps (25,000 and 350,000 with 5,000 and 70,000 as burn-in respectively), but matching the number of models evaluated. Results are presented in [Table 2](#). We report here the main findings:

- (i) over the 25 runs, ESS g with FSMH reaches the same top visited model 68% (17/25) while ESS g with MC³ the same top model only 28%, with a fixed τ , and 88% and 40% respectively with Z-S prior. This ability is extended to the top models ranked by the posterior probability, data not shown, providing indirect evidence that the proposed new move helps the algorithm to increase its predictive power. The great superiority when FSMH scheme is implemented can be explained by comparing subplot (a) and (c) in [Figure 1](#): the exchange of information between chains for ESS g with MC³ as local move when $p > n$ (and $p \gg p_\gamma$) is rather poor, negating the purpose of EMC. ESS g with MC³ has more difficulties to reach convergence in the product space and, in contrast to ESS g with FSMH, the retained chain does not easily escape from local modes. This later point can be seen looking at [Figure 1](#) (d) which magnifies the right hand tail of the kernel density of $\log p(\gamma|y)$ for the recorded chain, pulling together the 25 runs: interestingly ESS g with FSMH is less “bumpy”, showing a better ability to escape from local modes and to explore more efficiently the right tail.
- (ii) Regarding the second comparison ESS g with only block crossover overall beats constantly the other crossover operators in terms of best model visited ([Table 2](#)) and has higher acceptance rate ([Table 3](#)), showing also a great capacity to accumulate posterior mass as illustrated in [Figure 3](#).
- (iii) Regarding the hyperprior on τ , there is a better mixing ([Table 2](#)) and a higher acceptance rate of global moves when a random τ is implemented (see [Table 1](#) and [Table 3](#)). As a final comment we stress that besides improving mixing of the chains, a hyperprior on τ helps the search algorithm to focus in well supported models, as it is apparent in [Figure 2](#) (a).

	Version of ESSg	$\tau = 29$	$p(\tau)$
Experiment (i)	ESSg with only FSMH	68%	88%
	ESSg with only MC ³	28%	40%
Experiment (ii)	ESSg with only 1-point crossover	64%	80%
	ESSg with only block crossover	80%	84%
	ESSg with only uniform crossover	60%	84%
	ESSg with only adaptive crossover	60%	76%

Table 2: Proportion of times different versions of ESSg reach the same top visited model in the eQTL real data set with or without a hyperprior on τ in 25 replicates of the analysis.

	Version of ESSg	$\tau = 29$	$p(\tau)$
Experiment (ii)	ESSg with only 1-point crossover	0.303	0.335
	ESSg with only block crossover	0.482	0.501
	ESSg with only uniform crossover	0.026	0.042
	ESSg with only adaptive crossover	0	0.013

Table 3: Average acceptance rate of the crossover operator for different versions of ESSg in 25 replicates of the analysis of the first real data example, eQTL analysis, with or without a hyperprior on τ .

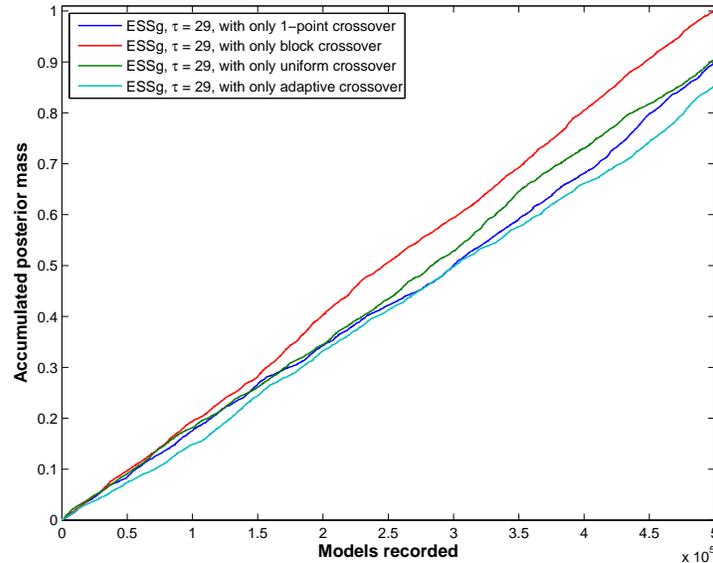


Figure 3: Accumulated posterior mass as a function of the models recorded. Plot generated using 25 replicates of the analysis of the first real data example, eQTL analysis, and normalised by the total mass found by ESSg, $\tau = 29$, with only block crossover move ($\rho_0 = 0.25$). 1-point and uniform crossover accumulate around 90% of the total mass accumulated by ESSg with only block crossover, while adaptive crossover only 85%.

4.2 Simulation study

We briefly report on a comprehensive study of the performance of ESS in a variety of simulated examples as well as a comparison with SSS. To make comparison with SSS fair, we use ESS*i*, the version of our algorithm which assumes independent priors, $\Sigma_\gamma = \tau I_{p_\gamma}$, with τ fixed at 1, the same prior setup specified in SSS for the regression coefficients. Details of the simulated examples (6 set-ups) and how we conducted the simulation experiment (25 replication of each set-up) are given in Appendix A.2. The rationale behind the construction of the examples was to benchmark our algorithm against both $n > p$ and $p > n$ cases, to use as building blocks intricate correlation structures that had been used in previous comparisons by [George and McCulloch \(1993, 1997\)](#) and [Nott and Green \(2004\)](#), as well as a realistic correlation structure derived from genetic data, and to include elements of model uncertainty in some of the examples by using a range of values of regression coefficients.

In our example we observe an effective exchange of information between the chains (reported in Table 4) which shows good overall acceptance rates the global moves that we have implemented. The dimension of the problem does not seem to affect the acceptance

rates in Table 4, remarkably since values of p range from 60 to 1,000 between the examples. We also studied specifically the performance of the global moves (Table 5) to scrutinise our temperature tuning and confirmed the good performance of ESS*i* with good frequencies of swapping (not far from the case where adjacent chains are selected to swap at random with equal probability) and good measures of overlap between chains.

In the same spirit of the real data example analysis, we also evaluate the superiority of the FSMH scheme with respect to more traditional EMC implementations, i.e when a MC³ local move is selected. While both versions of the search algorithm visit almost the same top models ranked by the posterior probability, ESS persists more on the top models.

All the examples were run in parallel with ESS*i* and SSS 2.0 (Hans et al. 2007) for the same number of sweeps (22,000) and matching hyperparameters on the model size. Comparison were made with respect to the marginal probability of inclusion as well as the ability to reach models with high posterior probability and to persist in that region. For a detailed discussion of all comparison, see Appendix A.2.3.

Overall the covariates with non-zero effects have high marginal posterior probability of inclusion for ESS*i* in all the examples, see Figure 4. There is good agreement between the two algorithms in general, with additional evidence on some examples (Figure 4 (c) and (d)) that ESS*i* is able to explore more fully the model space and in particular to find small effects, leading to a posterior model size that is close to the true one. Measures of goodness of fit and stability, Table 6, are in good agreement between ESS*i* and SSS. The comparison highlights that a key feature of SSS, its ability to move quickly towards the right model and to persist on it, is accompanied by a drawback in having difficulty to explore far apart models with competing explanatory power, in contrast to ESS*i* (contaminated example set-up). Altogether ESS*i* shows a small improvement of R^2_γ , related to its ability to pick up some of the small effects that are missed by SSS. Although it is difficult to compare the computational time between SSS and ESS*i* since they are written in two different languages, it seems that ESS*i* is less influenced by the number of simulated effects. To explore this point, we regressed the computational time in the 6 set-ups with respect to p and p_γ for both algorithms: while p has the same effect in the computational time for SSS and ESS*i*, the effect of p_γ in SSS is 30 times bigger than in ESS*i*. Altogether our comparisons show that we have designed a fully Bayesian MCMC-EMC sampler which is competitive with the effective search provided by SSS*i*.

	Ex1			Ex2	Ex3	Ex4	Ex5	Ex6
n	120			300	120	120	200	120
p	60			30	60	300	1,000	775
$E(p_\gamma)$	5	10	20	5	5	5	5	5
Crossover	0.249 (0.021)	0.270 (0.029)	0.271 (0.036)	0.157 (0.018)	0.215 (0.022)	0.147 (0.028)	0.170 (0.023)	0.193 (0.028)
DR Exch.	0.500 (0.040)	0.493 (0.043)	0.500 (0.040)	0.582 (0.020)	0.492 (0.071)	0.517 (0.105)	0.505 (0.013)	0.497 (0.072)

Table 4: Mean and standard deviation in brackets of EMC acceptance rates across replicates for ESS i with $\tau = 1$. “DR Exch.” stands for “delayed rejection exchange”.

	Ex1			Ex2	Ex3	Ex4	Ex5	Ex6	
n	120			300	120	120	200	120	
p	60			30	60	300	1,000	775	
$E(p_\gamma)$	5	10	20	5	5	5	5	5	
Swap.	$l = 1$	0.157	0.137	0.110	0.065	0.160	0.180	0.201	0.214
	$l = 2$	0.250	0.232	0.204	0.185	0.271	0.276	0.300	0.316
	$l = 3$	0.220	0.220	0.223	0.255	0.245	0.223	0.231	0.231
	$l = 4$	0.240	0.252	0.280	0.293	0.215	0.206	0.182	0.167
	$l = 5$	0.142	0.160	0.182	0.201	0.110	0.112	0.083	0.070
Overlap.	$l = 1, 2$	1.360	1.600	2.101	2.680	1.350	0.733	0.569	0.526
	$l = 2, 3$	1.570	1.570	1.600	0.870	1.430	1.021	0.913	0.893
	$l = 3, 4$	1.400	1.290	1.050	0.600	2.111	1.329	1.491	1.696
	$l = 4, 5$	1.100	0.992	0.690	1.251	4.131	1.503	2.304	2.499

Table 5: Swapping probability for ESS i with $\tau = 1$ defined as the observed frequency of successful swaps for each chain (including delayed rejection exchange and all-exchange operators) averaged across replicates. Overlapping measure defined as $V(f(\gamma_l))(1/t_{l+1} - 1/t_l)^2$, Liang and Wong (2000) with $f(\gamma_l) = \log p(y|\gamma_l) + \log p(\gamma_l)$. Target value for consecutive chains is $O(1)$.

		Ex1			Ex2	Ex3	Ex4	Ex5	Ex6
n		120			300	120	120	200	120
p		60			30	60	300	1,000	775
$E(p_\gamma)$		5	10	20	5	5	5	5	5
ESS i , $\tau=1$	R_γ^{2*}	0.864 (0.029)	0.867 (0.027)	0.871 (0.023)	0.975 (0.003)	≈ 1 (≈ 0)	0.962 (0.011)	0.703 (0.043)	0.997 (0.005)
	$\overline{R}_\gamma^{2**}$	0.863 (0.027)	0.866 (0.026)	0.874 (0.023)	0.975 (0.003)	≈ 1 (≈ 0)	0.957 (0.014)	0.689 (0.048)	0.997 (0.003)
	Stability	0.003 (0.001)	0.003 (0.002)	0.005 (0.002)	≈ 0 (≈ 0)	(≈ 0)	0.005 (0.004)	0.015 (0.007)	0.002 (0.002)
	Time (min.)	6 (< 1)	6 (< 1)	7 (< 1)	16 (< 1)	18 (1)	166 (32)	338 (43)	202 (40)
SSS	R_γ^{2*}	0.863 (0.027)	0.867 (0.025)	0.870 (0.024)	0.975 (0.003)	≈ 1 (≈ 0)	0.956 (0.016)	0.577 (0.074)	0.997 (0.004)
	$\overline{R}_\gamma^{2**}$	0.863 (0.027)	0.867 (0.025)	0.870 (0.024)	0.975 (0.003)	0.999 (≈ 0)	0.955 (0.016)	0.565 (0.078)	0.996 (0.004)
	Stability	0 (0)	0 (0)	≈ 0 (≈ 0)	≈ 0 (≈ 0)	≈ 0 (≈ 0)	0.001 (0.002)	0.009 (0.015)	0.004 (0.006)
	Time (min.)	12 (1)	12 (2)	13 (2)	118 (26)	497 (75)	502 (241)	169 (81)	549 (159)

Table 6: Comparison between ESS i with $\tau = 1$ and SSS for the six simulated examples. Standard deviation in brackets. R_γ^{2*} and $\overline{R}_\gamma^{2**}$ correspond to “ $R_\gamma^2: \max p(\gamma | y)$ ” and “ $\overline{R}_\gamma^2: 1,000$ largest $p(\gamma | y)$ ” respectively.

5 Discussion

The key idea in constructing an effective MCMC sampler for γ and τ is to add an extra parameter, the temperature, that weakens the likelihood contribution and enables escaping from local modes. Running parallel chains at different temperature is, on the other hand, expensive and the added computational cost has to be balanced against the gains arising from the various “exchanges” between the chains. This is why we focussed on developing a good strategy for selecting the pairs of chains, using both marginal and joint information between the chains, attempting bold and more conservative exchanges. Combining this with an automatic tuning of the temperature ladder during burn-in is one of the key element of our ESS algorithm. Using PT in this way has the potential to be effective in a wide range of situations where the posterior space is multimodal.

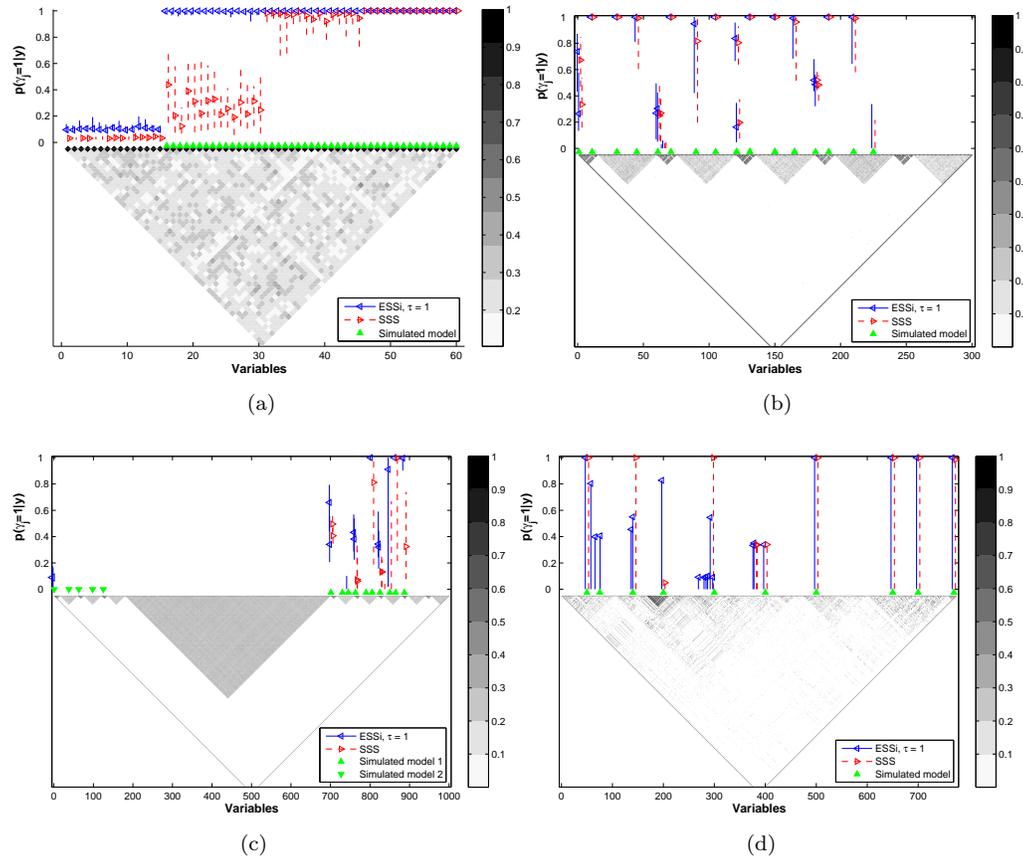


Figure 4: Median and interquartile range of the marginal posterior probability of inclusion (A.7) for Ex3, (a), Ex4, (b) and Ex5, (c), across replicates. Each graph is constructed as follows: bottom part, pairwise squared correlation $\rho^2(X_j, X_{j'})$, $j = 1, \dots, p$, between predictors for one selected replicate, grey scale indicates different values of squared correlation; blue left and red right triangles, median of $p(\gamma_j = 1 | y)$ across replicates for ESSi with $\tau = 1$ and SSS respectively; vertical blue solid lines and vertical red dashed lines, interquartile range of $p(\gamma_j = 1 | y)$ across replicates for ESSi and SSS respectively; upper and lower green triangles, simulated models. Selected replicate of Ex6, (d), shows marginal posterior probability of inclusion (blue left and red right triangles for ESSi $\tau = 1$ and SSS respectively). Marginal posterior probability of inclusion lower than 0.025 not shown.

When a model with a prior on the variable selection coefficient τ is preferred, the updating of τ itself present no particular difficulties and is computationally inexpensive. Moreover, using an adaptive sampler makes the algorithm self contained without any time consuming tuning of the proposal variance. This latter strategy works perfectly

well both in the g -prior and independent prior case as illustrated in Sections 4.1 and 4.2. Our current implementation does not make use of the output of the heated chains for posterior inference once the temperature placement is fixed at the end of the burn-in period. While it is not difficult to derive an overall marginal posterior probability of inclusion as the convex combination of the chain specific marginal probabilities of inclusion following the idea of Importance Tempering of Gramacy et al. (2010), investigating the potential gains in variance reduction that could be achieved is an area for further exploration, which is beyond the scope of the present work.

In summary in order to achieve a faithful exploration of the posterior model space in Bayesian variable selection, we recommend a parallel chain implementation with an automatic tuning of the temperature placement, FSMH as a local move and the all-exchange operator along with the block crossover as global moves. Moreover a hyperprior on τ is advisable in order to improve the mixing and the acceptance rate of the global moves as well as to discriminate among competing models.

Our approach has been applied so far to linear regression with univariate response y . An interesting generalisation is that of a multidimensional $n \times q$ response Y and the identification of regressors that jointly predict the Y Brown et al. (1998). Much of our set-up and algorithm carries through without difficulties and we have already implemented our algorithm in this framework in a challenging case study in genomics with multidimensional outcomes (Petretto et al. 2010).

Appendix

A.1 Technical details of EMC implementation

In this section we will describe some technical details omitted from the main text and related to the sampling schemes we used for the population of binary latent vectors γ and the selection coefficient τ .

A.1.1 EMC sampler for γ

Local move: FSMH scheme

Let $\gamma_{l,j}$, $l = 1, \dots, L$ and $j = 1, \dots, p$ to denote the j th latent binary indicator in the l th chain. As in Kohn et al. (2001), let $\gamma_{l,j}^{(1)} = (\gamma_{l,1}, \dots, \gamma_{l,j-1}, \gamma_{l,j} = 1, \gamma_{l,j+1}, \dots, \gamma_{l,p})^T$ and $\gamma_{l,j}^{(0)} = (\gamma_{l,1}, \dots, \gamma_{l,j-1}, \gamma_{l,j} = 0, \gamma_{l,j+1}, \dots, \gamma_{l,p})^T$. Furthermore let $L_{l,j}^{(1)} \propto p(y | \gamma_{l,j}^{(1)}, \tau)$ and $L_{l,j}^{(0)} \propto p(y | \gamma_{l,j}^{(0)}, \tau)$ and finally $\theta_{l,j}^{(1)} = p(\gamma_{l,j} = 1 | \gamma_{l,\setminus j})$ and $\theta_{l,j}^{(0)} = 1 - \theta_{l,j}^{(1)}$. From

(6) it is easy to prove that

$$\theta_{l,j}^{(1)} = p(\gamma_{l,j} = 1 | \gamma_{l,\setminus j}) = \frac{p_{\gamma_l} + a_\omega - 1}{p + a_\omega + b_\omega - 1}, \quad (\text{A.1})$$

where p_{γ_l} is the current model size for the l th chain. Using the above equation, for $\gamma_{l,j} = 1$ the normalised version of (12) can be written as

$$[p(\gamma_{l,j} = 1 | y, \gamma_{l,\setminus j}, \tau)]^{1/t_l} = \frac{\theta_{l,j}^{(1) 1/t_l} L_{l,j}^{(1) 1/t_l}}{S_{l,j}(1/t_l)}, \quad (\text{A.2})$$

where $S_{l,j}(1/t_l) = \theta_{l,j}^{(1) 1/t_l} L_{l,j}^{(1) 1/t_l} + \theta_{l,j}^{(0) 1/t_l} L_{l,j}^{(0) 1/t_l}$ with $[p(\gamma_{l,j} = 0 | y, \gamma_{l,\setminus j}, \tau)]^{1/t_l}$ defined similarly. Hence if $\theta_{l,j}^{(1) 1/t_l}$ is very small, then $[p(\gamma_{l,j} = 1 | y, \gamma_{l,\setminus j}, \tau)]^{1/t_l}$ is small as well. Therefore for the Gibbs sampler with a beta-binomial prior on the model space, the posterior probability of $\gamma_{l,j} = 1$ depends crucially on $\theta_{l,j}^{(1) 1/t_l}$.

In the following we derive a Fast Scan Metropolis-Hastings scheme specialised for Evolutionary Monte Carlo or parallel tempering. We define

$Q_{l,j}(1 \rightarrow 0) = Q(\gamma_{l,j}^{(1)} \rightarrow \gamma_{l,j}^{(0)})$ as the proposal probability to go from 1 to 0 and $Q_{l,j}(0 \rightarrow 1)$ the proposal probability to go from 0 to 1 for the j th variable in the l th chain. Moreover using the notation introduced before, the Metropolis-within-Gibbs version of (12) to go from 0 to 1 in the EMC local move is

$$\alpha_{i,j}^{\text{MwG}}(0 \rightarrow 1) = \min \left\{ 1, \frac{\theta_{l,j}^{(1) 1/t_l} L_{l,j}^{(1) 1/t_l} Q_{l,j}(1 \rightarrow 0)}{\theta_{l,j}^{(0) 1/t_l} L_{l,j}^{(0) 1/t_l} Q_{l,j}(0 \rightarrow 1)} \right\} \quad (\text{A.3})$$

with a similar expression for $\alpha_{i,j}^{\text{MwG}}(1 \rightarrow 0)$. We first introduce the following Proposition which is useful for the calculation of the acceptance probability in the FSMH scheme and to highlight the role of the prior density in (A.2). The proof is omitted since it is easy to check.

Proposition 1. *The following three conditions are equivalent: a) $L_{l,j}^{(1) 1/t_l} / L_{l,j}^{(0) 1/t_l} \geq 1$; b) $L_{l,j}^{(1) 1/t_l} / \tilde{S}_{l,j}(1/t_l) \geq 1$ where $\tilde{S}_{l,j}(1/t_l) = \tilde{\theta}_{l,j}^{(1)}(1/t_l) L_{l,j}^{(1) 1/t_l} + \tilde{\theta}_{l,j}^{(0)} L_{l,j}^{(0) 1/t_l}$ is the convex combination of the marginal likelihood $L_{l,j}^{(1) 1/t_l}$ and $L_{l,j}^{(0) 1/t_l}$ with weights $\tilde{\theta}_{l,j}^{(1)}(1/t_l) = \theta_{l,j}^{(1) 1/t_l} / (\theta_{l,j}^{(1) 1/t_l} + \theta_{l,j}^{(0) 1/t_l})$ and $\tilde{\theta}_{l,j}^{(0)}(1/t_l) = 1 - \tilde{\theta}_{l,j}^{(1)}(1/t_l)$; c) $L_{l,j}^{(0) 1/t_l} / \tilde{S}_{l,j}(1/t_l) < 1$.*

The FSMH scheme can be seen as a random scan Metropolis-within-Gibbs algorithm where the number of evaluations is linked to the prior/current model size and the temperature attached to the chain. The computation requirement for the additional acceptance/rejection step is very modest since the normalised tempered version of (A.1) is used.

Proposition 2. Let $l = 1, \dots, L$, $j = 1, \dots, p$ (or any permutation of them), $Q_{l,j}^{FSMH}(0 \rightarrow 1) = \tilde{\theta}_{l,j}^{(1)}(1/t_l)$ and $Q_{l,j}^{FSMH}(1 \rightarrow 0) = \tilde{\theta}_{l,j}^{(0)}(1/t_l)$ with $\tilde{\theta}_{l,j}^{(0)}(1/t_l) = 1 - \tilde{\theta}_{l,j}^{(1)}(1/t_l)$. The acceptance probabilities are

$$\alpha_{l,j}^{FSMH}(0 \rightarrow 1) = \begin{cases} 1 & \text{if } L_{l,j}^{(1)1/t_l} / L_{l,j}^{(0)1/t_l} \geq 1 \\ L_{l,j}^{(1)1/t_l} / L_{l,j}^{(0)1/t_l} & \text{if } L_{l,j}^{(1)1/t_l} / L_{l,j}^{(0)1/t_l} < 1 \end{cases} \quad (\text{A.4})$$

$$\alpha_{l,j}^{FSMH}(1 \rightarrow 0) = \begin{cases} 1 & \text{if } L_{l,j}^{(0)1/t_l} / L_{l,j}^{(1)1/t_l} \geq 1 \\ L_{l,j}^{(0)1/t_l} / L_{l,j}^{(1)1/t_l} & \text{if } L_{l,j}^{(0)1/t_l} / L_{l,j}^{(1)1/t_l} < 1 \end{cases} \quad (\text{A.5})$$

The above sampling scheme works as follows. Given the l th chain, if $\gamma_{lj} = 0$ (and similarly for $\gamma_{lj} = 1$), it proposes the new value from a Bernoulli distribution with probability $\tilde{\theta}_{l,j}^{(1)}(1/t_l)$: if the proposed value is different from the current one, it evaluates (A.4) (and similarly A.5) otherwise it selects a new covariate.

Finally it can be proved that the Gibbs sampler is more efficient than the FSMH scheme, i.e. for a fixed number of iterations, Gibbs sampling MCMC standard error is lower than for FSMH sampler. However the Gibbs sampler is computationally more expensive so that, if p is very large, as described in Kohn et al. (2001), FSMH scheme becomes more efficient per floating point operation.

Global move: exchange operator

The exchange operator can be seen as an extreme case of crossover operator, where the first proposed chain receives the whole second chain state $\gamma'_l = \gamma_r$, and the second proposed chain receives the whole first state chain $\gamma'_r = \gamma_l$, respectively.

In order to achieve a good acceptance rate, the exchange operator is usually applied on adjacent chains in the temperature ladder, which limits its capacity for mixing. To obtain better mixing, we implemented two different variations: the first one is based on Jasra et al. (2007) and the related idea of delayed rejection (Green and Mira 2001); the second one on Gibbs distribution over all possible chains pairs (Calvo 2005).

1. The delayed rejection exchange operator tries first to swap the state of the chains that are usually far apart in the temperature ladder, but, once the proposed move has been rejected, it performs a more traditional (uniform) adjacent pair selection, increasing the overall mixing between chains on one hand without drastically reducing the acceptance rate on the other. However its flexibility comes at some extra computational costs and in particular the additional evaluation of the pseudo move necessary to maintain detailed balance (Green and Mira 2001). Details are reported below.

Suppose two chains are selected at random, l and r with $l \neq r$, in order to swap their binary latent vector. Then, given that $\gamma'_l = \gamma_r$, $\gamma'_r = \gamma_l$ and $Q_t(\gamma \rightarrow \gamma') = Q_t(\gamma' \rightarrow \gamma)$, (13) reduces to

$$\alpha_1(\gamma \rightarrow \gamma') = \min \left\{ 1, \frac{\exp \{f(\gamma_r | \tau) / t_l + f(\gamma_l | \tau) / t_r\}}{\exp \{f(\gamma_l | \tau) / t_l + f(\gamma_r | \tau) / t_r\}} \right\}.$$

Since the two chains are selected at random, the above acceptance probability decreases exponentially with the difference $|1/t_l - 1/t_r|$ and therefore most of the proposed moves are rejected. If rejected, a delayed rejection-type move is applied between two random adjacent chains, with l the first one and s , $|l - s| = 1$, the second one, giving rise to the new acceptance probability

$$\alpha_2(\gamma \rightarrow \gamma'') = \min \left\{ 1, \frac{\exp \{f(\gamma_s | \tau) / t_l + f(\gamma_l | \tau) / t_s\} (1 - \alpha_1(\gamma'' \rightarrow \gamma^*))}{\exp \{f(\gamma_l | \tau) / t_l + f(\gamma_s | \tau) / t_s\} (1 - \alpha_1(\gamma \rightarrow \gamma'))} \right\},$$

where the pseudo move γ^* is necessary in order to maintain the detailed balance condition (Green and Mira 2001).

- Alternatively, we attempt a bolder “all-exchange” operator. Swapping the state of two chains that are far apart in the temperature ladder speeds up the convergence of the simulation since it replaces several adjacent swaps with a single move. However, this move can be seen as a rare event whose acceptance probability is low and unknown. Since the full set of possible exchange moves is finite and discrete, it is easy and computationally inexpensive to calculate all the $L(L-1)/2$ exchange acceptance rates between all chains pairs, inclusive the rare ones, $\tilde{p}_{l,r} = \exp \{(f(\gamma_r | \tau) - f(\gamma_l | \tau)) (1/t_l - 1/t_r)\}$. To maintain detailed balance condition, the possibility not to perform any exchange (rejection) must be added with un-normalised probability one. Finally the chains whose states are swapped are selected at random with probability equal to

$$p_h = \frac{\tilde{p}_h}{\sum_{h=1}^{1+L(L-1)/2} \tilde{p}_h}, \quad (\text{A.6})$$

where in (A.6) each pair $(l, r < l)$ is denoted by a single number h , $\tilde{p}_h = \tilde{p}_{l,r}$, including the rejection move, $h = 1$.

Temperature placement

First we select the number L of chains close to the complexity of the problem, i.e. $E(p_\gamma)$, although the size of the data and computational limits need to be taken into account. Secondly, we fix a first stage temperature ladder according to a geometric scale such that $t_{l+1}/t_l = b$, $b > 1$, $l = 1, \dots, L$ with b relatively large, for instance $b = 4$. Finally, we adopt a strategy similar to the one described in Roberts and Rosenthal (2009), but restricted to the burn-in stage, monitoring only the acceptance rate of the delayed rejection exchange operator. After the k th “batch” of EMC sweeps, to be chosen but usually set equal to 100, we update b_k , the value of the constant b up to

the k th batch, by adding or subtracting an amount δ_b such that the acceptance rate of the delayed rejection exchange operator is as close as possible to 0.50 (Liu 2001; Jasra et al. 2007) $b_{k+1} = 2^{\log_2 b_k \pm \delta_b}$. Specifically the value of δ_b is chosen such that at the end of the burn-in period the value of b can be 1. To be precise, we fix the value of δ_b as $\log_2(b_1)/\tilde{K}$, where b_1 is the first value assigned to the geometric ratio and \tilde{K} is the total number of batches in the burn-in period.

A.1.2 Adaptive Metropolis-within-Gibbs for τ

Since τ is defined on the real positive axis we propose the new value of τ on the logarithm scale. In particular we use as proposal the normal distribution centred at the current value of $\log(\tau)$ in the g -prior and independent prior case. The variance of the proposal distribution is controlled as illustrated in Roberts and Rosenthal (2009): every 100 EMC sweeps, the same value of sweeps used in the temperature placement, we monitor the acceptance rate of the Metropolis-within-Gibbs algorithm: if it is lower (higher) than the optimal acceptance rate, i.e. 0.44, a constant $\delta_\tau(k)$ is added (subtracted) to l_{s_k} , the log standard deviation of the proposal distribution in the k th batch of EMC sweeps. The value of the constant to be added or subtracted is rather arbitrary, but we found useful to fix it as $|l_{s_1} - 5|/\tilde{K}$, where \tilde{K} is the total number of batches in the burn-in period: during the burn-in the log standard deviation should be able to reach any values at a distance ± 5 in log scale from the initial value of l_{s_1} usually set equal to zero. The *diminishing adaptive condition* is obtained imposing $\delta_\tau(k) = \min\{|l_{s_1} - 5|/\tilde{K}, k^{-1/2}\}$, where k is the current number of batches, including the burn-in. To ensure the *bounded convergence condition* we follow Roberts and Rosenthal (2009), restricting each l_{s_k} to be inside $[M_1, M_2]$ and we fix them equal to $M_1 = -10$ and $M_2 = 10$ respectively. In practice these bounds do not create any restriction since the sequence of the standard deviations of the proposal distribution stabilises almost immediately, indicating that the transition kernel converges in a bounded number of batches.

A.2 Performance of ESS: Simulation study

In this section we report in details on the performance of ESS in a variety of simulated examples. Main conclusions are summarised in the Section 4.2.

Firstly we analyse the simulated examples with ESS*i* the version of our algorithm which assumes independent priors, $\Sigma_\gamma = \tau I_{p_\gamma}$, so as to enable comparisons with SSS which also implements an independent prior. Moreover, in order to make to comparison with SSS fair, in the simulation study only the first step of the algorithm described in Section 3.3 is performed, with τ fixed at 1. As in SSS, standardisation of the covariates is done before running ESS*i*. We run ESS*i* and SSS 2.0 (Hans et al. 2007) for the same number of sweeps (22,000) and with matching hyperparameters on the model size.

Secondly, to discuss the mixing properties of ESS when a prior $p(\tau)$ is defined on τ , we implement both the g -prior and independent prior set-up for a particular simulated experiment. To be precise in the former case we will use the Zellner-Siow priors (8), and

for the latter we will specify a proper but diffuse exponential distribution as suggested by [Bae and Mallick \(2004\)](#).

A.2.1 Simulated experiments

We apply ESS with independent priors to an extensive and challenging range of simulated examples with τ fixed at 1: the first three examples (Ex1-Ex3) consider the case $n > p$ while the remaining three (Ex4-Ex6) have $p > n$. Moreover in all examples, except the last one, we simulate the design matrix, creating more and more intricate correlation structures between the covariates in order to test the proposed algorithm in different and increasingly more realistic scenarios. In the last example, we use, as design matrix, a genetic region spanning 500-kb from the HapMap project ([Altshuler et al. 2005](#)).

Simulated experiments Ex1-Ex5 share in common the way we build X . In order to create moderate to strong correlation, we found useful referring to two simulated examples in [George and McCulloch \(1993, 1997\)](#): throughout we call X_1 ($n \times 60$) and X_2 ($n \times 15$) the design matrix obtained from these two examples. In particular the j th column of X_1 , indicated as $X_{(1)j}$, is simulated as $X_{(1)j} = X_j^* + Z$, where X_1^*, \dots, X_{60}^* iid $\sim N_n(0, 1)$ independently from $Z \sim N_n(0, 1)$, inducing a pairwise correlation of 0.5. X_2 is generated as follows: firstly we simulated Z_1, \dots, Z_{15} iid $\sim N_n(0, 1)$ and we set $X_{(2)j} = Z_i + 2Z_j$ for $j = 1, 3, 5, 8, 9, 10, 12, 13, 14, 15$ only. To induce strong multicollinearity, we then set $X_{(2)2} = X_{(2)1} + 0.15Z_2$, $X_{(2)4} = X_{(2)3} + 0.15Z_4$, $X_{(2)6} = X_{(2)5} + 0.15Z_6$, $X_{(2)7} = X_{(2)8} + X_{(2)9} - X_{(2)10} + 0.15Z_7$ and $X_{(2)11} = X_{(2)14} + X_{(2)15} - X_{(2)12} - X_{(2)13} + 0.15Z_{11}$. A pairwise correlation of about 0.998 between $X_{(2)j}$ and $X_{(2)j+1}$ for $j = 1, 3, 5$ is introduced and similarly strong linear relationship is present within the sets $(X_{(2)7}, X_{(2)8}, X_{(2)9}, X_{(2)10})$ and $(X_{(2)11}, X_{(2)12}, X_{(2)13}, X_{(2)14}, X_{(2)15})$.

Then, as in [Nott and Green \(2004\)](#) Example 2, more complex structures are created by placing side by side combinations of X_1 and/or X_2 , with different sample size. We will vary the number of samples n in X_1 and X_2 as we construct our examples. The levels of β are taken from the simulation study of [Fernández et al. \(2001\)](#), while the number of true effects, p_γ , with the exception of Ex3, varies from 5 to 16. Finally the simulated error variance ranges from 0.05^2 to 2.5^2 in order to vary the level of difficulty for the search algorithm. Throughout we only list the non-zero β_γ and assume that $\beta_{\setminus\gamma} = 0^T$. The six examples can be summarised as follows:

Ex1: $X = X_1$ is a matrix of dimension 120×60 , where the responses are simulated from (1) using $\alpha = 0$, $\gamma = (21, 37, 46, 53, 54)^T$, $\beta_\gamma = (2.5, 0.5, -1, 1.5, 0.5)^T$, and $\varepsilon \sim N(0, 2^2 I_{120})$. In the following we will not refer to the intercept α any more since, as described in Section 3.3 in the paper, we consider y centred and hence there is no difference in the results if the intercept is simulated or not. This is the simplest of our example, although, as reported in [George and McCulloch \(1993\)](#) the average pairwise correlation is about 0.5, making it already hard to analyse by standard stepwise methods.

- Ex2:** This example is taken directly from [Nott and Green \(2004\)](#), Example 2, who first introduce the idea of combining simpler “building blocks” to create a new matrix X : in their example $X = [X_2^{(1)} X_2^{(2)}]$ is a 300×30 matrix, where $X_2^{(1)}$ and $X_2^{(2)}$ are of dimension 300×15 and have each the same structure as X_2 . Moreover $\gamma = (1, 3, 5, 7, 8, 11, 12, 13)^T$, $\beta_\gamma = (1.5, 1.5, 1.5, 1.5, -1.5, 1.5, 1.5, 1.5)^T$ and $\varepsilon \sim N(0, 2.5^2 I_{300})$. We chose this example for two reasons: firstly, since the correlation structure in X_2 is very involved, we test the proposed algorithm under strong and complicated correlations between the covariates; secondly, since y is not simulated from the second “block”, we are interested to see if the proposed algorithm does *not* select any variable that belongs to the second group.
- Ex3:** As in [George and McCulloch \(1993\)](#), Example 2, $X = X_1$, is a 120×60 matrix, $\beta = (\beta_1, \dots, \beta_{60})^T$, $(\beta_1, \dots, \beta_{15}) = (0, \dots, 0)$, $(\beta_{16}, \dots, \beta_{30}) = (1, \dots, 1)$, $(\beta_{31}, \dots, \beta_{45}) = (2, \dots, 2)$, $(\beta_{46}, \dots, \beta_{60}) = (3, \dots, 3)$ and $\varepsilon \sim N(0, 2^2 I_{120})$. The motivation behind this example is to test the strength of the proposed algorithm to select a subset of variables which is large with respect to p while preserving the ability *not* to choose any of the first 15 variables.
- Ex4:** The design matrix X , 120×300 , is constructed as follows: firstly we create a new 120×60 “building block”, X_3 , combining X_2 and a smaller version of X_1 , X_1^* , a 120×45 matrix simulated as X_1 , such that $X_3 = [X_2 X_1^*]$ (dimension 120×60). Secondly we place side by side five copies of X_3 , $X = [X_3^{(1)} X_3^{(2)} X_3^{(3)} X_3^{(4)} X_3^{(5)}]$: the new design matrix alternates blocks of covariates of high and complicated correlation, as in [George and McCulloch \(1997\)](#), with regions where the correlation is moderate as in [George and McCulloch \(1993\)](#). We simulate the response selecting 16 variables from X ,
 $\gamma = (1, 11, 30, 45, 61, 71, 90, 105, 121, 131, 150, 165, 181, 191, 210, 225)^T$ such that every pair belongs alternatively to X_2 or X_1 . We simulate y using $\beta_\gamma = (2, -1, 1.5, 1, 0.5, 2, -1, 1.5, 1, 0.5, 2, -1, -1, 1.5, 1, 0.5)^T$ and $\varepsilon \sim N(0, 2.5^2 I_{120})$. This example is challenging in view of the correlation structure, the number of covariates $p > n$ and the different levels of the effects.
- Ex5:** This is the most challenging example and it is based on the idea of contaminated models. The matrix X , 200×1000 , is
 $X = [X_3^{(1)} X_3^{(2)} X_3^{(3)} X_1^{**} X_3^{(4)} X_3^{(5)} X_3^{(6)} X_3^{(7)} X_3^{(8)}]$, with X_1^{**} , a 200×520 larger version of X_1 . We partitioned the responses such that $y = [y_1^T y_2^T]^T$: y_1 is simulated from “model 1”
 $(\gamma^1 = (701, 730, 745, 763, 790, 805, 825, 850, 865, 887)$ and
 $\beta_\gamma^1 = (2, -1, 1.5, 1, 0.5, 2, -1, 1.5, 2, -1))$ while y_2 is simulated from “model 2”
 $(\gamma^2 = (1, 38, 63, 98, 125)$ and $\beta_\gamma^2 = (2, -1, 1.5, 1, 0.5))$. Finally, fixing
 $\varepsilon \sim N(0, 0.05^2 I_{200})$ and the sample size in the two models such that y_1 and y_2 are vectors of dimension 160×1 and 40×1 respectively, y is retained if, given the sampling variability, we find $R_{\gamma^1}^2 \geq 0.6$ and $R_{\gamma^1}^2/8 \leq R_{\gamma^2}^2 \leq R_{\gamma^1}^2/10$: in this way we know that “model 1” accounts for most of the variability of y , but

without a negligible effect for “model 2”. In this example, we measure the ability of the proposed algorithm to recognise the most promising model and therefore being robust to contaminations. However since ESS can easily jump between local modes we are also interested to see if “model 2” is selected.

Ex6: The last simulated example is based on phased genotype data from HapMap project (Altshuler et al. 2005), region ENm014, Yoruba population: the data set originally contained 1,218 SNPs (Single Nucleotide Polymorphism) for 120 chromosomes, but after eliminating redundant variables, the design matrix reduced to 120×775 . While in the previous examples a “block structure” of correlated variables is artificially constructed, in this example blocks of linkage disequilibrium (LD) derive naturally from genetic forces, with a slow decay of the level of pairwise correlation between SNPs. Finally we chose $\gamma = (50, 75, 140, 200, 300, 400, 500, 650, 700, 770)^T$ such that the effects are visually inside blocks of LD, with their size simulated from $\beta_\gamma \sim N(0, 3^2 I_{10})$ with $\varepsilon \sim N(0, 0.10^2 I_{120})$. Since the simulated effects can range roughly between $(-6, 6)$, this will allow us to test also the ability of ESS*i* to select small effects.

We conclude this section by reporting how we conducted the simulation experiment: every example from Ex1 to Ex6 has been replicated 25 times and the results presented for example Ex1 to Ex5 are averaged over the 25 replicates. For Ex6 the effects size change so average across replicated is only done for the mixing properties. ESS*i* with $\tau = 1$ was applied to each example/sample, recording the visited sequence of γ_1 for 20,000 sweeps after a burn-in of 2,000 required for the automatic tuning of the temperature placement, Section 3.1. We analysed all examples setting $E(p_\gamma) = 5$ with $V(p_\gamma) = E(p_\gamma)(1 - E(p_\gamma)/p)$ which corresponds to a binomial prior over p_γ . In order to establish the sensitivity of the proposed algorithm to the choice of $E(p_\gamma)$ we also analysed Ex1 fixing $E(p_\gamma) = 10$ and 20. Moreover in all the examples we chose $L = 5$ with the starting value of γ chosen at random. The remaining two hyperparameters to be fixed, namely a_σ and b_σ , are set equal to $a_\sigma = 10^{-10}$ and $b_\sigma = 10^{-3}$ as in Kohn et al. (2001) which corresponds to a relative uninformative prior.

A.2.2 Mixing properties of ESS*i*

In this section we report some stylised facts about the performance of the ESS*i* with τ fixed at 1. As expected, the chains attached to higher temperatures shows more variability. Albeit the convergence is reached in the product space $\prod_{l=1}^L [p(\gamma_l | y)]^{1/t_l}$, by visual inspection of the trace of $\log(p(\gamma_l | y))$ ($t_l = 1 \forall l, l = 1, \dots, L$) each chain *marginally* reaches its *equilibrium* with respect to the others; moreover, thanks to the automatic tuning of the temperature placement during the burn-in, the distributions of their log posterior probabilities overlap nicely, allowing effective exchange of information between the chains.

This effective exchange of information is demonstrated in Table 4 which shows good overall acceptance rates for the collection of moves that we have implemented. The dimension of the problem does not seem to affect the acceptance rate of the (delayed

rejection) exchange operator which stays very stable and close to the target: for instance in Ex4 ($p = 300$) and Ex6 ($p = 775$) the mean and standard deviation of the acceptance rate are 0.517 (0.105) and 0.497 (0.072) while in Ex5 ($p = 1,000$) we have 0.505 (0.013): the higher variability in Ex4 being related to the model size p_γ . With regards to the crossover operators, again we observe stability across all the examples. Moreover, in contrast to [Jasra et al. \(2007\)](#), when $p > n$, the crossover average acceptance rate across the five chains is quite stable between 0.147, Ex4, and 0.193, Ex6 (with the lower value in Ex4 here again due to p_γ): within our limited experiments, we believe that the good performance of crossover operator is related to the selection operator and the new block crossover, see Section 3.1.

Some finer tuning of the temperature ladder could still be performed as there seems to be an indication that fewer global moves are accepted with the higher temperature chain, see Table 5, where swapping probabilities for each chain are indicated. Note that the observed frequency of successful swaps is not far from the case where adjacent chains are selected to swap at random with equal probability. Other measures of overlapping between chains ([Liang and Wong 2000](#); [Iba 2001](#)), based on a suitable index of variation of $f(\gamma_t) = \log p(y|\gamma_t) + \log p(\gamma_t)$ across sweeps, confirm the good performance of ESS*i*. Again some instability is present in the high temperature chains, see in Table 5 the overlapping index between chains 3, 4 and 4, 5 in Example 3 to 6.

In Ex1, we also investigate the influence of different values of the prior mean of the model size. We found that the average (standard deviation in brackets) acceptance rate across replicates for the delayed rejection exchange operator ranges from 0.493 (0.043) to 0.500 (0.040) for different values of the prior mean on the model size, while the acceptance rate for the crossover operator ranges from 0.249 (0.021) to 0.271 (0.036). This strong stability is not surprising because the automatic tuning modifies the temperature ladder in order to compensate for $E(p_\gamma)$.

A.2.3 Performance of ESS*i* and comparison with SSS

Performance of ESS*i*

We conclude this section by discussing in details the overall performance of ESS*i* with respect to the selection of the true simulated effects. As a first measure of performance, we report for all the simulated examples the marginal posterior probability of inclusion as described in [George and McCulloch \(1997\)](#) and [Hans et al. \(2007\)](#). In the following, for ease of notation, we drop the chain subscript index and we exclusively refer to the first chain. To be precise, we evaluate the marginal posterior probability of inclusion as

$$p(\gamma_j = 1 | y) \simeq C^{-1} \sum_{t=1, \dots, T} 1_{(\gamma_j^{(t)}=1)}(\gamma) p(y | \gamma^{(t)}) p(\gamma^{(t)}) \tag{A.7}$$

with $C = \sum_{t=1, \dots, T} p(y | \gamma^{(t)}) p(\gamma^{(t)})$ and T the number of sweeps after the burn-in. The posterior model size is defined as $p(p_\gamma | y) \simeq C^{-1} \sum_{t=1, \dots, T} 1_{(|\gamma^{(t)}|=p_\gamma)}(\gamma) p(y | \gamma^{(t)}) p(\gamma^{(t)})$, with C as before. Besides

plotting the marginal posterior inclusion probability (A.7) averaged across sweeps and replicates for our simulated examples, we will also compute the interquartile range of (A.7) across replicates as a measure of variability.

In order to thoroughly compare the proposed ESS algorithm to SSS (Hans et al. 2007), we present also some other measures of performance based on $p(\gamma|y)$ and R_γ^2 : first we rank $p(\gamma|y)$ in decreasing order and record the indicator γ that corresponds to the maximum and 1,000 largest $p(\gamma|y)$ (after burn-in). Given the above set of latent binary vectors, we then compute the corresponding R_γ^2 leading to “ R_γ^2 : $\max p(\gamma|y)$ ” as well as the mean R_γ^2 over the 1,000 largest $p(\gamma|y)$, “ $\overline{R_\gamma^2}$: 1,000 largest $p(\gamma|y)$ ”, both quantities averaged across replicates. Moreover the actual ability of the algorithm to reach regions of high posterior probability and persist on them is monitored: given the sequence of the 1,000 best γ s (based on $p(\gamma|y)$), the standard deviation of the corresponding R_γ^2 s shows how stable is the searching strategy at least for the top ranked (not unique) posterior probabilities: averaging over the replicates, it provides an heuristic measures of “stability” of the algorithm. Finally we report the average computational time (in minutes) across replicates of ESS*i* written in Matlab code and run on a 2MHz CPU with 1.5 Gb RAM desktop computer and of SSS version 2.0 on the same computer.

Comparison with SSS

Figure 4 presents the marginal posterior probability of inclusion for ESS*i* with $\tau = 1$ averaged across replicates and, as a measure of variability, the interquartile range, blue left triangles and vertical blue solid line respectively. In general the covariates with non-zero effects have high marginal posterior probability of inclusion in all the examples: for example in Ex3, Figure 4 (a), the proposed ESS*i* algorithm, blue left triangle, is able to perfectly select the last 45 covariates, while the first 15, which do not contribute to y , receive small marginal posterior probability. It is interesting to note that this group of covariates, $(\beta_1, \dots, \beta_{15}) = (0, \dots, 0)$, although correctly recognised having no influence on y , show some variability across replicates, vertical blue solid line: however, this is not surprising since independent priors are less suitable in situations where all the covariates are mildly-strongly correlated as in this simulated example. On the other hand the second set of covariates with small effects, $(\beta_{16}, \dots, \beta_{30}) = (1, \dots, 1)$, are univocally detected. The ability of ESS*i* to select variables with small effects is also evident in Ex6, Figure 4 (d), where the two smallest coefficients, $\beta_2 = 0.112$ and $\beta_{10} = 0.950$ (the second and last respectively from left to right), receive from high to very high marginal posterior probability (and similarly for the other replicates, data not shown). In some cases however, some covariates attached with small effects are missed (e.g. Ex4, Figure 4 (b), the last simulated effect which is also the smallest, $\beta_{16} = 0.5$, is not detected). In this situation however the vertical blue solid line indicates that for some replicates, ESS*i* is able to assign small values of the marginal posterior probability giving evidence that ESS*i* fully explore the whole space of models.

Superimposed on all pictures of Figure 5 are the median and interquartile range across replicates of $p(\gamma_j = 1|y)$, $j = 1, \dots, p$, for SSS, red right triangles and vertical red dashed line respectively. We see that there is good agreement between the two

algorithms in general, with in addition evidence that *ESSi* is able to explore more fully the model space and in particular to find small effects, leading to a posterior model size that is close to the true one. For instance in Ex3, Figure 4 (a), where the last 30 covariates accounts for most of R_γ^2 , SSS has difficulty to detect $(\beta_{16}, \dots, \beta_{30})$, while in Ex6, it misses $\beta_2 = 0.112$, the smallest effect, and surprisingly also $\beta_4 = -2.595$ assigning a very small marginal posterior probability (and in general for the small effects in most replicates, data not shown). However the most marked difference between *ESSi* and SSS is present in Ex5: as for *ESSi*, SSS misses three effects of “model 1” but in addition $\beta_4 = 1$, $\beta_7 = -1$ and $\beta_8 = 1.5$ receive also very low marginal posterior probability, red right triangle, with high variability across replicates, vertical red dashed line. Moreover on the extreme left, as noted before, *ESSi* is able to capture the biggest coefficient of “model 2” while SSS misses completely all contaminated effects. No noticeable differences between *ESSi* and SSS are present in Ex1 and Ex2 for the marginal posterior probability, while in Ex4, SSS shows more variability in $p(\gamma_j = 1 | y)$ (red dashed vertical lines compared to blue solid vertical lines) for some covariates that do receive the highest marginal posterior probability.

In contrast to the differences in the marginal posterior probability of inclusion, there is general agreement between the two algorithms with respect to some measures of goodness of fit and stability, see Table 6. Again, not surprisingly, the main difference is seen in Ex5 where *ESSi* with $\tau = 1$ reaches a better R_γ^2 both for the maximum and the 1,000 largest $p(\gamma | y)$. SSS shows more stability in all examples, but the last: this was somehow expected since one key features of SSS in its ability to move quickly towards the right model and to persist on it (Hans et al. 2007), but a drawback of this is its difficulty to explore far apart models with competing R_γ^2 as in Ex5. Note that *ESSi* shows a small improvement of R_γ^2 in all the simulated examples. This is related to the ability of *ESSi* to pick up some of the small effects that are missed by SSS, see Figure 4. Finally *ESSi* seems to shows some superiority in terms of computational time (although comparing algorithms written in two different languages is difficult) especially when the simulated (and estimated) p_γ is large (in other simulated examples, data not shown, we found this is always true when $p_\gamma \gtrsim 10$): the explanation lies in the number of different models SSS and *ESSi* evaluate at each sweep. Indeed, SSS evaluates $p + p_\gamma(p - p_\gamma)$, where p_γ is the size of the current model, while *ESSi* theoretically analyses an equally large number of models, pL , but, when $p > n$, the actual number of models evaluated is drastically reduced thanks to our FSMH sampler. In only one case SSS beats *ESSi* in term of computational time (Ex5), but in this instance SSS clearly underestimates the simulated model and hence performs less evaluations than would be necessary to explore faithfully the model space. In conclusion, we see that the rich portfolio of moves and the use of parallel chains makes *ESS* robust for tackling complex covariate space as well as competitive against a state of the art search algorithm.

References

- Altshuler, D., Brooks, L. D., Chakravarti, A., Collins, F. S., Daly, M. D., and Donnelly, P. (2005). “A haplotype map of the human genome.” *Nature*, 437: 1299–1320. 610,

612

- Andersson, U., Lindberg, J., Wang, S., Marcusson-Staringhl, R. B. M., and et al. (2009). “A systems biology approach to understanding elevated serum alanine transaminase levels in a clinical trial with ximelagatran.” *Biomarkers*, 14: 572–586. 594
- Bae, N. and Mallick, B. K. (2004). “Gene selection using a two-level hierarchical Bayesian model.” *Bioinformatics*, 20: 3423–3430. 587, 592, 610
- Brown, P. J., Vannucci, M., and Fearn, T. (1998). “Multivariate Bayesian variable selection and prediction.” *Journal of the Royal Statistical Society - Series B*, 60: 627–641. 585, 588, 593, 605
- Calvo, F. (2005). “All-exchange parallel tempering.” *Journal of Chemical Physics*, 123: 1–7. 591, 607
- Chipman, H. (1996). “Bayesian variable selection with related predictors.” *Canadian Journal of Statistics*, 24: 17–36. 585
- Chipman, H., George, E. I., and McCulloch, R. E. (2001). “The practical implementation of Bayesian model selection (with discussion).” In Lahiri, P. (ed.), *Model Selection*, 66–134. IMS: Beachwood, OH. 583, 585, 586
- Clyde, M. A. and George, E. I. (2004). “Model uncertainty.” *Statistical Science*, 19: 81–94. 583, 585
- Cui, W. and George, E. I. (2008). “Empirical Bayes vs fully Bayes variable selection.” *Journal of Statistical Planning and Inference*, 138: 888–900. 586
- Dellaportas, P., Forster, J., and Ntzoufras, I. (2002). “On Bayesian model and variable selection using MCMC.” *Statistics and Computing*, 12: 27–36. 583, 589
- Dumas, M., Wilder, S. P., Bihoreau, M., Barton, R. H., Fearnside, J. F., and et al. (2007). “Direct quantitative trait locus mapping of mammalian metabolic phenotypes in diabetic and normoglycemic rat models.” *Nature Genetics*, 39: 666–672. 594
- Fernández, C., Ley, E., and Steel, M. F. J. (2001). “Benchmark priors for Bayesian model averaging.” *Journal of Econometrics*, 75: 317–343. 583, 585, 586, 594, 610
- George, E. I. and McCulloch, R. E. (1993). “Variable selection via Gibbs sampling.” *Journal of the American Statistical Association*, 88: 881–889. 589, 600, 610, 611
- (1997). “Approaches for Bayesian variable selection.” *Statistica Sinica*, 7: 339–373. 600, 610, 611, 613
- Goswami, G. and Liu, J. S. (2007). “On learning strategies for evolutionary Monte Carlo.” *Statistics and Computing*, 17: 23–38. 591
- Gramacy, R. B., Samworth, R. J., and King, R. (2010). “Importance Tempering.” *Statistics and Computing*, 20: 1–7. 605

- Green, P. J. and Mira, A. (2001). “Delayed rejection in reversible jump Metropolis-Hastings.” *Biometrika*, 88: 1035–1–53. 591, 607, 608
- Hans, C., Dobra, A., and West, M. (2007). “Shotgun Stochastic Search for “large p ” regression.” *Journal of the American Statistical Association*, 102: 507–517. 583, 584, 587, 601, 609, 613, 614, 615
- Hübner, N., Wallace, C. A., Zimdahl, H., Petretto, E., Schulz, H., and et al. (2005). “Integrated transcriptional profiling and linkage analysis for identification of genes underlying disease.” *Nature Genetics*, 37: 243–253. 593
- Iba, Y. (2001). “Extended Ensemble Monte Carlo.” *International Journal of Modern Physics C*, 12: 623–656. 594, 613
- Jasra, A., Stephens, D. A., and Holmes, C. (2007). “Population-based reversible jump Markov chain Monte Carlo.” *Biometrika*, 94: 787–807. 583, 587, 588, 591, 594, 607, 609, 613
- Kindmark, A., Jawaid, A., Harbron, C. G., Barrat, B. J., and March, R. E. (2008). “Genome-wide pharmacogenetic investigation of a hepatic adverse event without clinical signs of immunopathology suggests an underlying immune pathogenesis.” *Pharmacogenomics Journal*, 8: 186–195. 594
- Kohn, R., Smith, M., and Chan, D. (2001). “Nonparametric regression using linear combinations of basis functions.” *Statistics and Computing*, 11: 313–322. 583, 585, 586, 589, 605, 607, 612
- 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*, 481: 410–423. 583, 586
- Liang, F. and Wong, W. H. (2000). “Evolutionary Monte Carlo: application to C_p model sampling and change point problem.” *Statistica Sinica*, 10: 317–342. 583, 587, 590, 598, 602, 613
- Liu, J. (2001). *Monte Carlo strategies in scientific computations*. Springer: New York. 591, 609
- Madigan, D. and York, J. (1995). “Bayesian graphical models for discrete data.” *International Statistical Review*, 63: 215–232. 588
- Maruyama, Y. and George, I. E. (2008). “A g -prior extension for $p > n$.” Technical report.
URL <http://arxiv.org/abs/0801.4410v1> 586
- Natarajan, R. and McCulloch, C. E. (1998). “Gibbs sampling with diffuse proper priors: a valid approach to data-driven inference.” *Journal of Computational and Graphical Statistics*, 7: 267–277. 592

- Nott, D. J. and Green, P. J. (2004). “Bayesian variable selection and the Swendsen-Wang algorithm.” *Journal of Computational and Graphical Statistics*, 13: 141–157. 587, 600, 610, 611
- O’Hara, R. B. and Sillanpää, M. J. (2009). “A review of Bayesian variable selection methods: what, how and which.” *Bayesian Analysis*, 4: 85–118. 583
- Petretto, E., Bottolo, L., Langley, S. R., Heinig, M., McDermott-Roe, M. C., Sarwar, R., Pravenec, M., Hübner, N., Aitman, T. J., Cook, S. A., and Richardson, S. (2010). “New insights into the genetic control of gene expression using a Bayesian multi-tissue approach.” *PLoS Computational Biology*, 6: e1000737. 605
- Roberts, G. O. and Rosenthal, J. S. (2009). “Examples of adaptive MCMC.” *Journal of Computational and Graphical Statistics*, 9: 349–367. 592, 593, 608, 609
- Tierney, L. and Kadane, J. B. (1986). “Accurate approximations for posterior moments and marginal densities.” *Journal of the American Statistical Association*, 81: 82–86. 591
- Wilson, M. A., Iversen, E. S., Clyde, M. A., Schmidler, S. C., and Shildkraut, J. M. (2009). “Bayesian model search and multilevel inference for SNP association studies.” Technical report.
URL <http://arxiv.org/abs/0908.1144> 583
- Zellner, A. (1986). “On assessing prior distributions and Bayesian regression analysis with g-prior distributions.” In Goel, P. K. and Zellner, A. (eds.), *Bayesian Inference and Decision Techniques-Essays in Honour of Bruno de Finetti*, 233–243. North-Holland: Amsterdam. 585
- Zellner, A. and Siow, A. (1980). “Posterior odds ratios for selected regression hypotheses.” In Bernardo, J. M., DeGroot, M. H., Lindley, D. V., and Smith, A. F. M. (eds.), *Bayesian Statistics, Proc. 1st Int. Meeting*, 585–603. University of Valencia Press: Valencia. 586

Acknowledgments

The authors are thankful to Norbert Hübner and Timothy Aitman for providing the data of the eQTL example, Enrico Petretto for the interpretation of the results in the real data examples, Gareth Roberts and Jeffrey Rosenthal for helpful discussions about adaptation and Michail Papatomas for his detailed comments. We are also grateful to the the editor, associate editor and two anonymous referees for valuable comments that improved the presentation of the paper. Sylvia Richardson acknowledges support from the MRC grant G060020609.