Genet. Sel. Evol. 35 (2003) 585–604

DOI: 10.1051/gse:2003041


Original article

# A comparison of alternative methods to compute conditional genotype probabilities for genetic evaluation with finite locus models

Liviu R. Totir[a*], Rohan L. Fernando[a,b], Jack C.M. Dekkers[a,b], Soledad A. Fernández[c], Bernt Guldbrandtsen[d]

[a] Department of Animal Science, Iowa State University, Ames, IA 50011-3150, USA
[b] Lawrence H. Baker Center for Bio-informatics and Biological Statistics, Iowa State University, Ames, IA 50011-3150, USA
[c] Department of Statistics, The Ohio State University, Columbus, OH 43210, USA
[d] Danish Institute of Animal Science, Foulum, Denmark

(Received 27 February 2002; accepted 5 May 2003)

**Abstract –** An increased availability of genotypes at marker loci has prompted the development of models that include the effect of individual genes. Selection based on these models is known as marker-assisted selection (MAS). MAS is known to be efficient especially for traits that have low heritability and non-additive gene action. BLUP methodology under non-additive gene action is not feasible for large inbred or crossbred pedigrees. It is easy to incorporate non-additive gene action in a finite locus model. Under such a model, the unobservable genotypic values can be predicted using the conditional mean of the genotypic values given the data. To compute this conditional mean, conditional genotype probabilities must be computed. In this study these probabilities were computed using iterative peeling, and three Markov chain Monte Carlo (MCMC) methods — scalar Gibbs, blocking Gibbs, and a sampler that combines the Elston Stewart algorithm with iterative peeling (ESIP). The performance of these four methods was assessed using simulated data. For pedigrees with loops, iterative peeling fails to provide accurate genotype probability estimates for some pedigree members. Also, computing time is exponentially related to the number of loci in the model. For MCMC methods, a linear relationship can be maintained by sampling genotypes one locus at a time. Out of the three MCMC methods considered, ESIP, performed the best while scalar Gibbs performed the worst.

**genotype probabilities / finite locus models / Markov chain Monte Carlo**

* Corresponding author: ltotir@iastate.edu

## 1. INTRODUCTION

Marker assisted genetic evaluation (MAGE) is most useful for traits with low heritability [23,27] that exhibit non-additive gene action [6]. Under non-additive inheritance, however, BLUP is difficult to implement, especially when inbreeding is present [7]. To overcome the computing problems associated with BLUP under non-additive gene action, it has been proposed to predict the unobservable genotypic values using the conditional mean of the genotypic values given the data, calculated under the assumption of a finite locus model [14,19,28]. Furthermore, crossbred data do not increase the complexity of this type of prediction. The conditional mean of the genotypic values given the data is also known as the best predictor (BP) because, conditional on the assumed model being correct, it minimizes the mean square error of prediction, and selection using BP maximizes the mean genotypic value of the selected candidates [4,13]. The appropriateness of finite locus models for genetic evaluation for quantitative traits is currently under investigation, and preliminary results indicate that models with 2–10 loci yield evaluations that are practically indistinguishable from BLUP evaluations [30,31].

In the frequentist approach to BP, the conditional genotypic values are computed from the true values of the model parameters and genotype probabilities conditional on the data and on the true values of the model parameters. In practice, however, the true values of the model parameters are not known. Thus, estimates of the model parameters are used in place of the true values. In the Bayesian approach, the conditional genotypic values are obtained by marginalizing over the unknown parameter values [17]. In practice, marginalizing the unknown parameters is done using Markov chain Monte Carlo (MCMC) methods. This Bayesian approach will usually require computing genotype probabilities conditional on the data and on specified values of the model parameters. Thus, both approaches will require an efficient method to compute conditional genotype probabilities. Under a finite locus model, these probabilities can be calculated exactly by the Elston-Stewart algorithm [9], approximated by iterative peeling [11,32], or estimated by MCMC methods [14,19,28].

The Elston-Stewart algorithm is computationally practicable only for simple pedigrees [15], and for models with no more than about three loci. Iterative peeling can be applied to large pedigrees, but it yields exact probabilities only for pedigrees without loops [15,33]. The performance of iterative peeling for computing conditional genotype probabilities under finite locus models with more than one locus has not been studied. Janss *et al.* [21] studied the potential of using the Gibbs sampler to analyze quantitative traits in animal genetics. They found that the scalar Gibbs sampler has mixing problems in pedigrees that contain large sibships. This is due to the dependence between the genotypes of parents and offspring [21]. Scalar Gibbs is, however, still

one of the most widely used MCMC methods for genetic analyses [1,8,24,25]. Blocking Gibbs was recommended as an alternative to scalar Gibbs in order to overcome the dependence problem [21]. The blocking scheme suggested by Janss *et al.* [21], samples the genotype of a sire jointly with the genotypes of its terminal offspring. A more extreme alternative is to use peeling and reverse peeling to sample jointly the genotypes of all animals in a pedigree [11,20]. This strategy, however, is not feasible when the pedigree contains many nested loops. For such pedigrees, an approximate method has been proposed in order to obtain candidate samples and accept or reject these by the Metropolis-Hastings algorithm [11,20]. An MCMC sampler called ESIP combines the Elston-Stewart algorithm with iterative peeling to obtain candidate samples from the entire pedigree; these samples are then accepted or rejected using a Metropolis-Hastings algorithm [11].

In order to further study the potential of finite locus models for genetic evaluation of quantitative traits, a reliable method is required to efficiently compute conditional genotype probabilities given the data. Thus, the objective of this paper was to study the performance of iterative peeling, scalar Gibbs, blocking Gibbs, and ESIP when used to calculate conditional genotype probabilities for a quantitative trait in finite locus models. Simulated data were used to assess the performance of the methods by calculating BP given the true values of the model parameter.

## 2. METHODS

Consider a trait determined by $N$ segregating quantitative trait loci (QTL) with two alleles at each locus. For a population of $n$ individuals, a given genotypic configuration of this trait can be written as a matrix $\boldsymbol{G}$ of dimension $n \times N$

$$\boldsymbol{G} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1N} \\ g_{21} & g_{22} & \cdots & g_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ g_{n1} & g_{n2} & \cdots & g_{nN} \end{bmatrix}, \tag{1}$$

where $g_{ij}$ denotes the genotype of individual $i$ at locus $j$. $\boldsymbol{G}$ can also be written as

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \vdots \\ \boldsymbol{g}_i \\ \vdots \\ \boldsymbol{g}_n \end{bmatrix}, \tag{2}$$

where $\boldsymbol{g}_i$ is the $1 \times N$ vector of genotypes of individual $i$, or as

$$\boldsymbol{G} = \left[\boldsymbol{c}_1\ \boldsymbol{c}_2\ \ldots\ \boldsymbol{c}_j\ \ldots\ \boldsymbol{c}_N\right], \tag{3}$$

where $\boldsymbol{c}_j$ is the $n \times 1$ column vector of genotypes at locus $j$. When only additive and dominance gene actions are present, following Bulmer [4], the vector $\boldsymbol{v}$ of genotypic values of $n$ individuals can be modeled as

$$\begin{aligned}\boldsymbol{v} &= \mathbf{1}\eta + \sum_{j=1}^{N} \boldsymbol{v}_j \\ &= \mathbf{1}\eta + \sum_{j=1}^{N} \boldsymbol{Q}_j \boldsymbol{\delta}_j,\end{aligned} \tag{4}$$

where $\mathbf{1}$ is a $n \times 1$ vector of ones; $\eta$ is the trait mean [10]; $\boldsymbol{v}_j$ is the $n \times 1$ vector of genotypic values at locus $j$ deviated from the trait mean; $\boldsymbol{Q}_j$ is an $n \times 3$ incidence matrix relating the genotypic deviations at locus $j$ to the corresponding individuals, with each row $\boldsymbol{q}_{ij}$ of $\boldsymbol{Q}_j$ being one of the vectors [1 0 0], [0 1 0], or [0 0 1]; and $\boldsymbol{\delta}_j$ is a $3 \times 1$ vector that contains the genotypic effects at locus $j$: $[a_j\ d_j\ -a_j]'$ [10]. The vector $\boldsymbol{y}$ of phenotypic values of $n$ individuals under a finite locus model can be written as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}(\mathbf{1}\eta + \boldsymbol{Q}\boldsymbol{\delta}) + \boldsymbol{e}, \tag{5}$$

where $\boldsymbol{X}$ is the incidence matrix relating the vector $\boldsymbol{\beta}$ of fixed effects to $\boldsymbol{y}$; $\boldsymbol{Z}$ is the incidence matrix relating $\boldsymbol{v}$ to $\boldsymbol{y}$; $\boldsymbol{Q} = [\boldsymbol{Q}_1\ \boldsymbol{Q}_2\ \ldots\ \boldsymbol{Q}_N]$; $\boldsymbol{\delta} = [\boldsymbol{\delta}_1\ \boldsymbol{\delta}_2\ \ldots\ \boldsymbol{\delta}_N]'$; $\boldsymbol{e}$ is the vector of residuals. The parameters of this model are: $\boldsymbol{\beta}$, $\eta$, the genotypic values $a_j$ and $d_j$, and gene frequency $p_j$ for locus $j = 1, \ldots, N$, and the residual variance $\sigma^2$. In this paper, we assumed all parameters are known. The only unknowns are the genotypes at the $N$ loci.

The conditional mean of the vector of genotypic values given phenotypic values, which is also the best predictor (*BP*), can be written as

$$\mathrm{E}(\boldsymbol{v} \mid \boldsymbol{y}) = \mathbf{1}\eta + \sum_{\boldsymbol{G}} \boldsymbol{v}_{\boldsymbol{G}} \Pr(\boldsymbol{G} \mid \boldsymbol{y}), \tag{6}$$

where $\boldsymbol{v}_{\boldsymbol{G}}$ is the vector of genotypic deviations that corresponds to the genotypic configuration $\boldsymbol{G}$, and

$$\Pr(\boldsymbol{G} \mid \boldsymbol{y}) = \frac{f(\boldsymbol{G}, \boldsymbol{y})}{f(\boldsymbol{y})} \propto f(\boldsymbol{y} \mid \boldsymbol{G}) \Pr(\boldsymbol{G}), \tag{7}$$

where $f(\boldsymbol{y} \mid \boldsymbol{G})$ is the conditional probability density function of the phenotypic values given $\boldsymbol{G}$, and $\Pr(\boldsymbol{G})$ is the probability of the genotype configuration $\boldsymbol{G}$.

Under a finite locus model, the phenotypic values are assumed to be independent given the genotypes. As a result we can write

$$f(\mathbf{y} \mid \boldsymbol{G}) = \prod_{i=1}^{n} f(y_i \mid \boldsymbol{g}_i), \tag{8}$$

where $f(y_i \mid \boldsymbol{g}_i)$ is the conditional probability density function of phenotype $y_i$ given that individual $i$ has genotype $\boldsymbol{g}_i$. This conditional probability density function is also known as the penetrance function [16]. If individuals are numbered such that ancestors precede descendants, and if the founder genotypes are assumed to be independent, the probability of a given genotypic configuration can be written as

$$\Pr(\boldsymbol{G}) = \prod_{i \in F} \Pr(\boldsymbol{g}_i) \prod_{i \in C} \Pr(\boldsymbol{g}_i \mid \boldsymbol{g}_{mi}, \boldsymbol{g}_{fi}), \tag{9}$$

where $F$ is the set of founder individuals and $C$ is the set of nonfounders. For $i \in F$, the probability of the vector $\boldsymbol{g}_i$ of genotypes for individual $i$ can be written as

$$\Pr(\boldsymbol{g}_i) = \prod_{j=1}^{N} \Pr(g_{ij}), \tag{10}$$

where $\Pr(g_{ij})$ is equal to the population frequency of $g_{ij}$. Assuming the QTL are unlinked, for $i \in C$ the conditional probability that offspring $i$ will have the genotype vector $\boldsymbol{g}_i$ given the parents of $i$ have the genotype vectors $\boldsymbol{g}_{mi}$ and $\boldsymbol{g}_{fi}$ can be written as

$$\Pr(\boldsymbol{g}_i \mid \boldsymbol{g}_{mi}, \boldsymbol{g}_{fi}) = \prod_{j=1}^{N} \Pr(g_{ij} \mid g_{mij}, g_{fij}), \tag{11}$$

where $\Pr(g_{ij} \mid g_{mij}, g_{fij})$ is the conditional probability that offspring $i$ will have the genotype $g_{ij}$ at locus $j$ given that the parents of $i$ have the genotypes $g_{mij}$ and $g_{fij}$ at locus $j$ [2,9].

The key problem in any implementation of genetic evaluation using a finite locus model is the correct and efficient calculation of the sum over all possible genotypic configurations ($\boldsymbol{G}$) in equation (6). The following methods were used here: the Elston-Stewart algorithm, iterative peeling, and three different MCMC methods (scalar Gibbs, blocking Gibbs, and ESIP).

### 2.1. Elston-Stewart algorithm

For simple pedigrees and models with up to three loci, the Elston-Stewart algorithm [9] can be used to efficiently compute the sum over all genotypic configurations and obtain exact genetic evaluations. These exact genetic evaluations were used here as reference values to assess the performance of the four methods under investigation.

### 2.2. Iterative peeling

Iterative peeling applied to pedigrees has been discussed by several authors [15,32,33]. When pedigrees have loops, iterative peeling results in an extended pedigree [33]. Fernandez *et al.* [11] describe iterative peeling using directed graphs to represent pedigrees. They provide general expressions that allow the use of iterative peeling in arbitrary directed graphs. Fernandez *et al.* [11] implemented iterative peeling for the analysis of phenotypic data of a biallelic disease locus. For this type of inheritance, the genotype completely determines the phenotype, and thus, the penetrance function is a simple indicator function. For the purpose of this paper, we used the approach of Fernandez *et al.* [11], but for models with different numbers of independent loci. For these models, the calculation of transition probabilities was done as shown in equation (11). Also, for these type of models, the penetrance function $f(y_i \mid \boldsymbol{g}_i)$ is given by the density function of a normal distribution with mean $\eta + \sum_j \boldsymbol{q}_{ij}\boldsymbol{\delta}_j$ and variance $\sigma^2$.

### 2.3. MCMC methods

#### *2.3.1. General considerations*

Monte Carlo integration can be used to estimate expectations of random variables [18]. The BP can be estimated by simple Monte Carlo integration if we can draw independent samples from $\Pr(\boldsymbol{G} \mid \boldsymbol{y})$. In most cases, however, it is not feasible to draw independent samples from this distribution. It is often feasible to generate samples from a Markov chain with $\Pr(\boldsymbol{G} \mid \boldsymbol{y})$ as its stationary distribution. Monte Carlo integration using samples from a Markov chain is called MCMC. All three MCMC methods under investigation (scalar Gibbs, blocking Gibbs, and ESIP) give accurate results if the Markov chains are sufficiently long. The efficiency of these methods is characterized by the computing time needed to obtain accurate results. Various convergence diagnostics are used to determine the length required for accurate results [3,18]. However, none of the available convergence diagnostics is foolproof [3,18]. For all the situations considered in this paper, the exact evaluations of BP can be calculated by the Elston-Stewart algorithm. Thus, we did not need to rely on convergence diagnostics to determine the length of the chain required to obtain accurate results.

For each of the three MCMC methods under investigation, an initial sample from $\Pr(\boldsymbol{G} \mid \boldsymbol{y})$ was needed. To obtain this, the genotypes of the ancestors were sampled before those of the descendants. For founders, genotypes were sampled using the cumulative distribution function (cdf) of $(\boldsymbol{g}_i \mid y_i)$. For nonfounders, genotypes were sampled using the cdf of $(\boldsymbol{g}_i \mid \boldsymbol{g}_{mi}, \boldsymbol{g}_{fi}, y_i)$. Once an initial sample was obtained, new genotype samples were generated one locus at a time conditional on the genotypes at all the other loci. Before moving to the

next locus, genotypes were sampled within the current locus for all individuals. The three MCMC methods differ in the way the genotypes are sampled within a locus.

### *2.3.2. Scalar Gibbs*

For scalar Gibbs, each $g_{ij}$ is sampled conditional on $\boldsymbol{y}$ and all the other genotypes ($\boldsymbol{G}_{ij-}$). Due to the Markovian nature of the genetic data, however, the genotype of an individual is completely determined by the genotypes of the individuals that form its neighborhood: parents, mates, and descendants. As a result, the genotype $g_{ij}^t$ of nonfounder $i$ at locus $j$ in step $t$ was sampled from

$$\Pr(g_{ij} \mid \boldsymbol{y}, \boldsymbol{G}_{ij-}^t) = \frac{\Pr(g_{ij} \mid g_{mij}^t, g_{fij}^t) f(y_i \mid \boldsymbol{g}_i^t) \prod_{k \in O_i} \Pr(g_{kj}^t \mid g_{ij}, g_{o_kj}^t)}{\sum_{g_{ij}} \text{numerator}}, \tag{12}$$

where $g_{mij}^t$ and $g_{fij}^t$ represent the current genotypes of the parents of $i$;

$$\boldsymbol{g}_i^t = [g_{i1}^t \; g_{i2}^t \cdots g_{ij-1}^t \; g_{ij} \; g_{ij+1}^{t-1} \cdots g_{iN}^{t-1}]; \tag{13}$$

$O_i$ is the set of offspring of $i$; $g_{kj}^t$ is the current genotype of offspring $k$ at locus $j$; $g_{o_kj}^t$ is the current genotype of the other parent of $k$ at locus $j$. For founders the same formula was used except that $\Pr(g_{ij} \mid g_{mij}^t, g_{fij}^t)$ was replaced by $\Pr(g_{ij})$. This sampling process is repeated for all individuals within locus $j$. Once all individuals were sampled within locus $j$, the same process was repeated for locus $j + 1$.

### *2.3.3. Blocking Gibbs*

For blocking Gibbs, genotypes at locus $j$ were sampled using the blocking scheme suggested by Janss *et al.* [21], where the genotypes of sires and their terminal offspring are sampled jointly. For sire $i$ with a set $T_i$ of terminal offspring, $g_{ij}$ was sampled conditional on $\boldsymbol{y}$ and all other genotypes except the genotypes at locus $j$ for the terminal offspring ($\boldsymbol{G}_{ij,T_ij-}$). Thus, the genotype $g_{ij}^t$ of a nonfounder sire $i$ at locus $j$ in step $t$ was sampled from

$$\Pr(g_{ij} \mid \boldsymbol{y}, \boldsymbol{G}_{ij,T_ij-}^t) = \frac{\Pr(g_{ij} \mid g_{mij}^t, g_{fij}^t) f(y_i \mid \boldsymbol{g}_i^t) \prod_{k \in N_i} \Pr(g_{kj}^t \mid g_{ij}, g_{o_kj}^t) \prod_{l \in T_i} \sum_{g_{lj}} \Pr(g_{lj} \mid g_{ij}, g_{o_lj}^t) f(y_l \mid \boldsymbol{g}_l^t)}{\sum_{g_{ij}} \text{numerator}}, \tag{14}$$

where $N_i$ is the set of non terminal offspring of $i$; $g_{o_kj}^t$ is the current genotype of the other parent of $k$ at locus $j$; $g_{o_lj}^t$ is the current genotype of the other parent of $l$ at locus $j$;

$$\boldsymbol{g}_l^t = [g_{l1}^t \; g_{l2}^t \cdots g_{lj-1}^t \; g_{lj} \; g_{lj+1}^{t-1} \cdots g_{lN}^{t-1}]. \tag{15}$$

For founder sires the same formula was used except that $\Pr(g_{ij} \mid g^t_{mij}, g^t_{fij})$ is replaced with $\Pr(g_{ij})$. For terminal offspring $l$ of sire $i$, $g^t_{lj}$ was sampled from the cdf of $(g_{lj} \mid g^t_{ij}, g^t_{olj}, y_l)$. For other individuals, $g^t_{ij}$ was sampled according to (12). Once all individuals were sampled within locus $j$, the same process was repeated for locus $j+1$.

### *2.3.4. ESIP*

For ESIP, genotypes at locus $j$ were sampled as described by Fernandez *et al.* [11], where joint genotype samples from the entire pedigree are obtained by reverse peeling [11,20]. For example, a sample in step $t$ is obtained by sampling sequentially

$$\begin{aligned}
&g^t_{1j} \quad \text{from} \quad \Pr(g_{1j} \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-}),\\
&g^t_{2j} \quad \text{from} \quad \Pr(g_{2j} \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-}, g^t_{1j}),\\
&g^t_{3j} \quad \text{from} \quad \Pr(g_{3j} \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-}, g^t_{1j}, g^t_{2j}),\\
&\vdots\\
&g^t_{nj} \quad \text{from} \quad \Pr(g_{nj} \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-}, g^t_{1j}, g^t_{2j}, g^t_{3j} \ldots, g^t_{n-1j}),
\end{aligned} \tag{16}$$

where $\boldsymbol{G}^t_{j-} = \left[\boldsymbol{c}^t_1 \ldots \boldsymbol{c}^t_{j-1}\ \boldsymbol{c}^{t-1}_{j+1} \ldots \boldsymbol{c}^{t-1}_N\right]$ is the current genotype configuration at all the other loci except locus $j$ at step $t$. Note that the resulting sample comes from $\Pr(g_{1j}, g_{2j}, g_{3j} \ldots, g_{nj} \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-}) = \Pr(\boldsymbol{c}_j \mid \boldsymbol{y}, \boldsymbol{G}^t_{j-})$, where $\boldsymbol{c}_j$ is the genotype configuration at locus $j$. The Elston-Stewart algorithm can be used to calculate the probabilities needed in the sampling process [5,9]. In the Elston-Stewart algorithm, intermediate results must be stored in multidimensional tables called cutsets [11]. For pedigrees without loops, only two-dimensional tables are generated. For pedigrees with many nested loops, the dimension of the cutsets may increase to the point that the Elston-Stewart algorithm may not be feasible anymore. As a result, the Elston-Stewart algorithm cannot be used for this type of pedigrees. Fernandez *et al.* [11] have combined the Elston-Stewart algorithm with iterative peeling to make the joint sampling of genotypes feasible for arbitrary pedigrees. In this combined approach, the Elston-Stewart algorithm is used while the cutset size is small enough, and iterative peeling is used for the remainder of the pedigree. It can be shown that the results from the iterative peeling are equivalent to those obtained by the Elston-Stewart algorithm for a modified pedigree [33]. Candidate samples from a modified pedigree were generated by using the combined approach. These candidate samples were then accepted or rejected through a Metropolis-Hastings algorithm. The Metropolis-Hastings algorithm used corresponded to the special case of independence sampling [11]. For this case, the acceptance probability of a move from the

genotype configuration $\boldsymbol{c}_j^{t-1}$ to genotype configuration $\boldsymbol{c}_j^t$ is given by

$$\alpha(\boldsymbol{c}_j^{t-1}, \boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) = \min\left(1, \frac{\pi(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) \times q(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t)}{\pi(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t) \times q(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t)}\right), \tag{17}$$

where

$$\pi(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) = \Pr(\boldsymbol{c}_j^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \tag{18}$$

is the target probability of the genotype configuration $\boldsymbol{c}_j^t$,

$$\pi(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t) = \Pr(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \tag{19}$$

is the target probability of the genotype configuration $\boldsymbol{c}_j^{t-1}$,

$$q(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) = \Pr_{\mathrm{M}}(\boldsymbol{c}_j^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \tag{20}$$

is the probability of the candidate sample, where the subscript M is used to denote that, if iterative peeling is used, this sample is drawn from a modified pedigree. Finally,

$$q(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t) = \Pr_{\mathrm{M}}(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \tag{21}$$

is the probability of $\boldsymbol{c}_j^{t-1}$, if $\boldsymbol{c}_j^{t-1}$ would be sampled from the same distribution as $\boldsymbol{c}_j^t$. The target probability of genotype configuration $\boldsymbol{c}_j^t$, for example, was calculated as follows

$$\pi(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) \propto \prod_{i \in F} \Pr(g_{ij}^t) f(y_i \mid \boldsymbol{g}_i^t) \prod_{i \in C} \Pr(g_{ij}^t \mid g_{mij}^t, g_{fij}^t) f(y_i \mid \boldsymbol{g}_i^t). \tag{22}$$

Next consider the calculation of $q(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t)$. This can be done as follows

$$\begin{aligned} q(\boldsymbol{c}_j^t \mid \boldsymbol{G}_{j-}^t) = {} & \Pr_{\mathrm{M}}(g_{1j}^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \times \Pr_{\mathrm{M}}(g_{2j}^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^t) \\ & \times \Pr_{\mathrm{M}}(g_{3j}^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^t, g_{2j}^t) \times \cdots \\ & \times \Pr_{\mathrm{M}}(g_{nj}^t \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^t, g_{2j}^t, g_{3j}^t \ldots, g_{n-1j}^t), \end{aligned} \tag{23}$$

where $g_{ij}^t$ denotes the genotype sampled for animal $i$ at locus $j$ in step $t$. Note that all probabilities that form the product in equation (23) were already calculated in the reverse peeling process used to sample $\boldsymbol{c}_j^t$. Now consider the calculation of $q(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t)$. This is not as straightforward because $\boldsymbol{c}_j^{t-1}$ was sampled from $\Pr_{\mathrm{M}}(\boldsymbol{c}_j \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^{t-1})$, while what we needed to calculate was $q(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t)$. This probability can be calculated as follows

$$\begin{aligned} q(\boldsymbol{c}_j^{t-1} \mid \boldsymbol{G}_{j-}^t) = {} & \Pr_{\mathrm{M}}(g_{1j}^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t) \times \Pr_{\mathrm{M}}(g_{2j}^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^{t-1}) \\ & \times \Pr_{\mathrm{M}}(g_{3j}^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^{t-1}, g_{2j}^{t-1}) \times \cdots \\ & \times \Pr_{\mathrm{M}}(g_{nj}^{t-1} \mid \boldsymbol{y}, \boldsymbol{G}_{j-}^t, g_{1j}^{t-1}, g_{2j}^{t-1}, g_{3j}^{t-1} \ldots, g_{n-1j}^{t-1}), \end{aligned} \tag{24}$$

where $g_{ij}^{t-1}$ denotes the genotype sampled for animal $i$ at locus $j$ in step $t-1$. The probabilities that form the left-hand side product in equation (24) were calculated using the same intermediate results from the Elston-Stewart algorithm that were used to calculate the probabilities that form the left-hand side product of equation (23).

Finally, note that if only the Elston-Stewart algorithm is used to calculate the probabilities needed in the sampling process, $q$ is the same as $\pi$, and as a result all samples are accepted.

### 2.4. Simulation study

Three hypothetical pedigrees were used to assess the performance of the four methods under investigation. The first hypothetical pedigree is shown in Figure 1.

This pedigree had 96 individuals, several loops, and each of its nuclear families had 10 offspring. This pedigree will be referred to as the base pedigree. The second pedigree is an extension of the base pedigree. The extension was done by assigning to individuals 66, 67, 87, 77, 56 the same parental role as that of individuals 1, 2, 3, 14, 15, and then duplicating the structure of the base pedigree for three more generations. As a result, the second pedigree had seven generations and 187 individuals and will be referred to as the extended pedigree. Finally, a third pedigree with a family structure typical for a poultry population was considered. This pedigree consisted of one male mated to eight females with each mating producing 15 offspring. It had 129 individuals and no loops and will be referred to as the poultry pedigree.

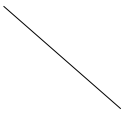

**Figure 1.** Base Pedigree.

**Table I.** Situations simulated. No. missing denotes the number of parents with missing phenotypic information. $h_n^2$ denotes the narrow sense heritability and $h_b^2$ denotes the broad sense heritability. No. samples denotes the number of samples generated with ESIP.

| Situation | Pedigree | No. loci | No. missing | $h_n^2$ | $h_b^2$ | No. samples |
|---|---|---|---|---|---|---|
| 1 | base | 1 | 15 | 0.04 | 0.08 | 75 000 |
| 2 | base | 1 | 15 | 0.4 | 0.8 | 3500 |
| 3 | base | 2 | 15 | 0.04 | 0.08 | 195 000 |
| 4 | base | 2 | 15 | 0.4 | 0.8 | 250 000 |
| 5 | base | 2 | 0 | 0.04 | 0.08 | 180 000 |
| 6 | extended | 2 | 15 | 0.04 | 0.08 | 175 000 |
| 7 | poultry | 2 | 9 | 0.04 | 0.08 | 175 000 |
| 8 | poultry | 3 | 9 | 0.04 | 0.08 | 230 000 |

In order to examine the effect of pedigree structure, missing data, number of loci in the model, and genetic parameters on the accuracy of genetic evaluations, eight situations were considered (Tab. I).

For each situation, ten replicates of the pedigree phenotypes were generated. For each situation, the simulation model and the analysis models were identical. The simulation study was designed so that the Elston-Stewart algorithm could be used to obtain exact genetic evaluations for each situation considered. All loci of a given finite locus model had the same parameters. Thus, all loci had equal gene frequencies and additive and dominance effects. Situation 3 was used as the reference situation in the design of the simulation study. The genetic parameters for this situation were similar to estimates reported in the animal science literature for low heritable traits that exhibit non-additive gene action [6]. For this situation, all parents in the base pedigree (15 individuals) were assumed to have missing phenotype information.

The first four situations of Table I were designed to consider all possible combinations of two heritabilities (0.04 and 0.4) and two values for the the number of loci in the model (one and two). This design allowed us to examine the main effects of heritability and number of loci in the model, as well as the effect of their interaction, for the base pedigree. Situation 5, which differs from situation 3 only in the number of missing phenotypes, was considered to examine the effect of missing data. Situations 6 and 7, which differ from situation 3 only in the pedigree structure, were considered to examine the effect of the pedigree. Situation 8, which differs from situation 7 only in the number of loci, was considered to examine the effect of the number of loci in the poultry pedigree. For the base and extended pedigree, only the models with one or two loci were considered due to the computational limitations of the Elston-Stewart algorithm.

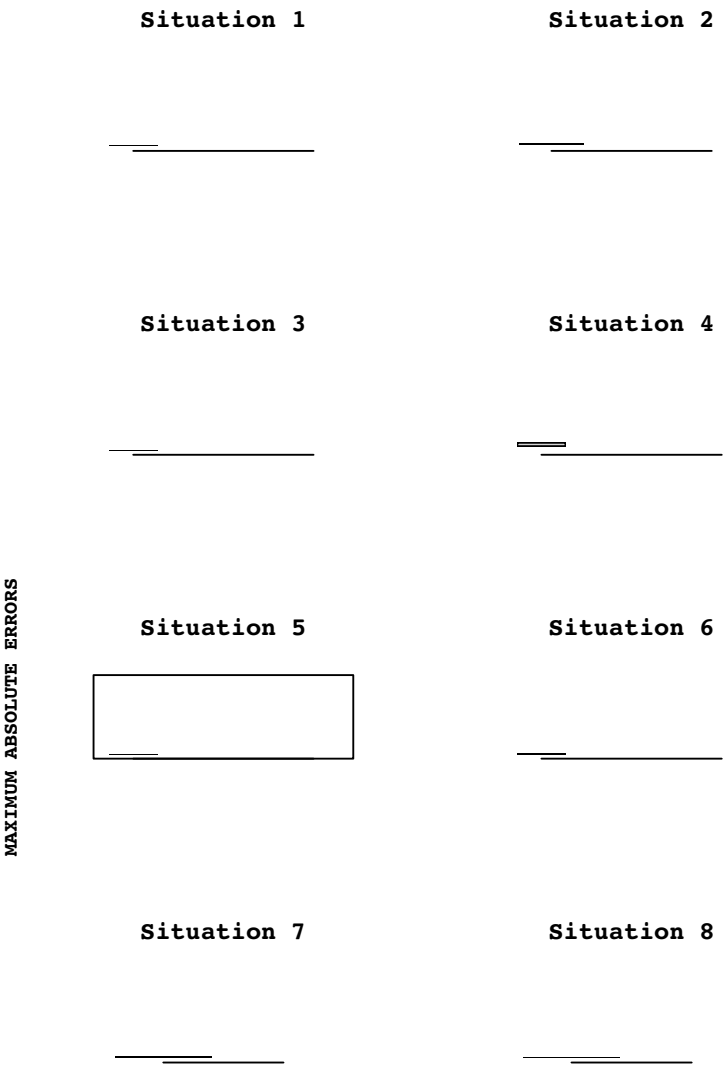


**Figure 2.** Box plots for the maximum absolute errors generated by ESIP, blocking Gibbs (BG), scalar Gibbs (SG), and iterative peeling (IP) for each of the situations 1–8.

Equation (6) was used to obtain estimates of genotypic values. In (6), the sum over the possible genotypic configurations was calculated exactly when the Elston-Stewart algorithm was used. When iterative peeling was used, the sum was calculated exactly for pedigrees without loops and approximated for pedigrees with loops. Finally, when the MCMC methods were used, the sum was estimated by sampling.

For each individual, the scaled absolute difference between the genetic evaluation obtained with each of the four methods under investigation (iterative peeling, scalar Gibbs, blocking Gibbs, and ESIP) and the exact evaluation obtained with the Elston-Stewart algorithm was calculated. The scaling factor used was the genetic standard deviation for each situation considered. These scaled absolute differences will be referred to as absolute errors. Even if a method yields accurate evaluations for the majority of the candidates for selection, the presence of a large absolute error for some individuals would make such a method unsuitable for genetic evaluation. Thus, in order to study the accuracy of the four methods used for genetic evaluation, the maximum of the absolute errors was computed for each replicate. As a result, for a given situation, each of the four methods generated ten maximum absolute errors. Figure 2 summarizes these values for each of the eight situations in the form of box plots.

A box plot is a graphical representation of a distribution [26]. The lower edge of the gray box represents the 25th percentile, the line within the gray box the 50th percentile, and the upper edge the 75th percentile. The lower and the upper whiskers represent the minimum and the maximum. By visual inspection of these figures, we can make statistical inferences about the performance of the four methods. This graphical method of inference is preferred to an analysis of variance because of the large heterogeneity of residual variances across methods (see Fig. 2).

Estimates obtained using MCMC methods depend on the number of samples used to calculate them. To make a fair comparison between the three MCMC methods, equal computing time was allocated to each method. The mean sum of the squares of the unscaled absolute differences was used as the convergence criterion. In the first replicate of each situation, the ESIP sampler was run until the convergence criterion was less than or equal to 0.0001 (Tab. I). The same amount of computing time as used in the first replicate of a given situation was then used for any other MCMC run under that situation.

## 3. RESULTS

### 3.1. Iterative peeling

Five iterations were used to obtain approximate genetic evaluations by iterative peeling. The effect of a larger number of iterations on the accuracy of genetic evaluations was negligible. Fernandez *et al.* [11] showed that iterative peeling yields very good approximations for conditional genotype probabilities in the case of a recessive disease trait. For the one-locus models considered in our study (situations 1 and 2), Figure 2 indicates that for quantitative traits iterative peeling can yield absolute errors that are

larger than 0.1 genetic standard deviations. For some parents these absolute errors were as high as 0.39 genetic standard deviations. Figure 2 also shows that the variability of the maximum absolute errors for iterative peeling was higher for high heritability (situations 2 and 4) than for low heritability (situations 1 and 3). The approximations obtained for two locus models (situations 3 and 4) were similar to those obtained for one-locus models (situations 1 and 2).

For the base pedigree, missing phenotypic records had almost no impact, as seen by comparing the box plot of situation 3 with the box plot of situation 5. Iterative peeling performed worst for the extended pedigree of situation 6, which has a larger number of loops. Iterative peeling yielded exact results for situations 7 and 8 because the poultry pedigree has no loops, and thus was not represented in Figure 2.

### 3.2. Influence of the number of loci on computing efficiency

As described below, the exponential relationship between computing efficiency and the number of loci in the model restricts the practical use of iterative peeling to models with about three loci. With iterative peeling, genotype probabilities must be calculated for every multilocus genotype. Given two alleles at each locus, the number of possible genotypes is $3^N$. Iterative peeling involves working with a three-dimensional table of conditional probabilities for the genotype of an offspring given the genotypes of its parents. Thus the number of computations required is proportional to

$$\left(3^N\right)^3 \times n \times i, \tag{25}$$

where $i$ is the number of iterations. In contrast, when MCMC samplers are used, a linear relationship between computing efficiency and the number of loci in the model can be maintained by sampling genotypes one locus at a time. Table II reflects this linear relationship for each of the three MCMC samplers under investigation.

**Table II.** Computing time in seconds on a Dec Alphastation 500 for 1000 samples obtained with each of the three MCMC samplers for 1 2 and 3 locus models for situation 1.

| Sampler | No. of loci | | |
|---|---|---|---|
| | 1 | 2 | 3 |
| ESIP | 83 | 166 | 249 |
| Blocking Gibbs | 12 | 24 | 36 |
| Scalar Gibbs | 6 | 12 | 18 |

### 3.3. MCMC methods

#### *3.3.1. Mixing behavior of MCMC samplers*

In order to investigate the mixing behavior of the three MCMC samplers, the mean and the standard error (S.E.) of the convergence criterion was calculated across the ten replicates of each of the eight situations at several stages of each MCMC sampler. Plots of the mean minus $3 \times$ S.E. and the mean plus $3 \times$ S.E. across all stages of the three MCMC samplers were then used to visually inspect the behavior of each MCMC sampler. Except for situation 4, the mean of the convergence criterion was the lowest for ESIP at all stages of a run. For situation 4, all three samplers reached a high level of accuracy in a short period of time.

#### *3.3.2. ESIP*

Because ESIP was used as the reference sampler, the accuracy of ESIP estimates were similar for all situations. It is of interest, however, to examine the difference in the number of samples needed to reach the desired level of accuracy for the eight situations considered (Tab. I). In general, all things being equal, as the amount of genetic information increased, the number of samples needed decreased. For example, situations 1 and 2 differed only in the heritability of the traits modeled. Situation 2, which corresponds to a highly heritable trait, needed a smaller number of samples compared with situation 1, which corresponds to a lowly heritable trait. For a highly heritable trait, the distribution of the genotypic values given the phenotypes is narrow. As a result, a small number of samples was needed to obtain accurate estimates for the conditional mean of the genotypic values given the phenotypes. To reach the same level of accuracy for a lowly heritable trait, however, a larger number of samples was needed, because now the distribution of the genotypic values given the phenotypes is more dispersed. Situations 3 and 4, however, contradicted this pattern. Situation 4, which corresponds to a highly heritable trait, needed a larger number of samples compared with situation 3, which corresponds to a lowly heritable trait. For these two situations, however, a two-locus model was used. The high number of samples needed in situation 4 indicated the presence of a mixing problem. This type of behavior has been reported when sampling tightly linked loci, and has been referred to as horizontal dependence [29]. Although in this paper the trait loci were unlinked, horizontal dependence was generated through the penetrance function when sampling one locus at a time and when heritability was high. Consider, for example, the genotypes $\begin{bmatrix}0 & 1\end{bmatrix}$ and $\begin{bmatrix}1 & 0\end{bmatrix}$. If the two loci that form each genotype vector have equal gene frequencies and genotypic effects, the two genotypes will have equal genotypic values. As a result, these two genotypes should be sampled in equal proportions given the data. When sampling genotypes one locus at a time, however,

it is not possible to move from $\boldsymbol{g}_i^t = \begin{bmatrix} 0 & 1 \end{bmatrix}$ to $\boldsymbol{g}_i^{t+k} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ in one step (*i.e.*, $k = 1$). An intermediate step through either genotype $\boldsymbol{g}_i^{t+k'} = \begin{bmatrix} 0 & 0 \end{bmatrix}$ or genotype $\boldsymbol{g}_i^{t+k'} = \begin{bmatrix} 1 & 1 \end{bmatrix}$, where $k' < k$, needs to occur first. The genotypic values of $\begin{bmatrix} 0 & 0 \end{bmatrix}$ and $\begin{bmatrix} 1 & 1 \end{bmatrix}$ are different from the genotypic value of $\begin{bmatrix} 0 & 1 \end{bmatrix}$ and $\begin{bmatrix} 1 & 0 \end{bmatrix}$. For a trait with low heritability the penetrance function is dispersed. This generates overlaps for different genotypic values. Consequently, the required intermediate move from $\begin{bmatrix} 0 & 1 \end{bmatrix}$ to $\begin{bmatrix} 0 & 0 \end{bmatrix}$ or $\begin{bmatrix} 1 & 1 \end{bmatrix}$ is more likely.

The difference in the number of samples needed in situation 1 *versus* situation 3, or 7 *versus* 8, emphasizes a second effect caused by the increase in the number of loci in the model. As the number of loci increased, the number of samples needed to reach the same level of accuracy increased as well because of the larger number of genotype probabilities that needed to be estimated. For practical purposes, however, the loss in accuracy due to horizontal dependence and the number of genotype probabilities to be estimated was negligible, because ESIP reached a high level of accuracy very fast.

#### *3.3.3. Blocking Gibbs*

Except for situation 4, blocking Gibbs yielded estimates that were significantly less accurate than the estimates obtained by ESIP (Fig. 2). In these situations, the absolute errors for some individuals were between 0.1 and 0.39 genetic standard deviations. For situation 4, blocking Gibbs reached almost the same level of accuracy as ESIP (Fig. 2).

#### *3.3.4. Scalar Gibbs*

For situation 1, scalar Gibbs had almost the same accuracy as blocking Gibbs but was significantly less accurate than ESIP (Fig. 2). For situation 2, scalar Gibbs exhibited poor mixing, with some replicates yielding absolute errors of up to 2.6 genetic standard deviations, and thus the box plot for this situation was not included in Figure 2. Note that the only difference between situations 1 and 2 was the heritability of the trait. The low heritability in situation 1 helped overcome the mixing problem due to the vertical dependence between parents and offspring. The results for situations 3 and 4 were similar to those obtained with blocking Gibbs (Fig. 2). The mixing problem observed in situation 2 disappeared in situation 4, where a two-locus model was used. In this case, the benefit of breaking the vertical dependence by increasing the number of loci outweighed the loss in accuracy caused by the introduction of horizontal dependence. For situation 5, the results were again similar to those obtained with blocking Gibbs (Fig. 2). The extension of the base pedigree in situation 6 increased the vertical dependence between parents and offspring. For this situation, a slight loss in accuracy was observed when compared with the level of accuracy reached for situation 3. Slow mixing was very severe for situations 7

and 8, situations with strong vertical dependence generated by the large number of offspring per parent. For the poultry pedigree, neither low heritability nor an increase in the number of loci (two and three, respectively) could alleviate the mixing problem generated by the vertical dependence between parents and offspring. Again no box plots were generated because some of the absolute errors were as large as 3.2 genetic standard deviations.

### 3.4. Implementation of ESIP

The results presented so far for ESIP were obtained by only using the Elston-Stewart algorithm. Thus, all proposed samples were accepted. The Elston-Stewart algorithm can be used as long as the cutset size is not too large for efficient computations. Once the cutset size becomes too large, iterative peeling is used and the proposed samples come from a modified pedigree. As a result, some of the proposed samples will be rejected. However, for the situations considered, even when iterative peeling was used, ESIP with 50 000 samples yielded more accurate results in a fraction of the computing time than scalar Gibbs and blocking Gibbs with a much larger number of samples.

## 4. DISCUSSION

Iterative peeling yielded exact results for pedigrees without loops regardless of the number of loci considered. For pedigrees with loops, the accuracy of the approximations obtained by iterative peeling decreased as the number of loops increased. Besides the limited accuracy for pedigrees with loops, iterative peeling has a serious limitation due to the exponential relationship between computing time and the number of loci in the model. However, a linear relationship between computing efficiency and the number of loci can be maintained for MCMC methods by sampling one locus at a time.

Out of the three MCMC methods considered, scalar Gibbs had the poorest performance overall because of poor mixing due to vertical dependence between parents and offspring. Although this problem has been recognized in the early stages of the development of MCMC methods, scalar Gibbs is still widely used because it is easy to implement and because of its per-sample computational efficiency. Joint updating of genotypes has been proposed to overcome this problem [22]. The blocking Gibbs sampler implemented in this paper, jointly updates the genotype of a sire and the genotypes of its terminal offspring within each locus. The ESIP sampler, however, jointly updates all genotypes within each locus. However, joint updating reduces the per-sample computational efficiency. The results of this paper show that, given equal computing time, blocking Gibbs and ESIP, which used joint updating, outperformed scalar Gibbs in terms of accuracy of the genetic evaluations. Furthermore, ESIP,

which jointly updated all genotypes within a locus, reached a higher level of accuracy than the other two samplers in a fraction of the computing time. In this paper we have established ESIP as an efficient method for calculating conditional genotype probabilities in finite locus models. Further studies are required to investigate the impact of unknown model parameter values on genetic evaluation with finite locus models.

Throughout this paper BP were obtained for the genotypic value as opposed to obtaining separate BP for the additive and the dominance components of the genotypic value. As explained below, under dominance inheritance, when inbreeding or cross-breeding is practiced, the additive genotypic value of an animal is not a good indicator of the performance of future offspring. Under additive inheritance, the additive genotypic value of a future offspring is equal to the mean additive genotypic values of the parents. Under dominance inheritance, when inbreeding or cross-breeding is practiced, the genotypic value of a future offspring is not equal to the additive genotypic values of the parents. For example, when there is overdominance, the additive covariance between parent and offspring can be negative [12]. Thus, in this situation parents can be selected based on the BP of the genotypic values of future offspring.

## ACKNOWLEDGEMENTS

This journal paper of the Iowa Agriculture and Home Economics Experiment Station, Ames, Iowa, Project No. 6587, was supported by the Hatch Act and State of Iowa funds, and was partially funded by award No. 2002-35205-1156 of the National Research Initiative Competitive Grants Program of the USDA. The helpful comments from an anonymous reviewer are gratefully acknowledged.

## REFERENCES

[1] Bink M.C.A.M., van Arendonk J.A.M., Quaas R.L., Breeding value estimation with incomplete marker data, Genet. Sel. Evol. 30 (1998) 45–58.
[2] Bonney G.E., On the statistical determination of major gene mechanisms in continuous human traits: regressive models, Am. J. Med. Genet. 18 (1984) 731–749.
[3] Brooks S.P., Gelman A., General methods for monitoring convergence of iterative simulations, Comp. Graph. Stat. 7 (1998) 434–455.
[4] Bulmer M.G., The mathematical theory of quantitative genetics, Clarendon Press, Oxford, 1980.
[5] Cannings C., Thompson E.A., Skolnick M.H., Probability functions on complex pedigrees, Adv. Appl. Prob. 10 (1978) 26–61.
[6] Culbertson M.S., Mabry J.W., Misztal I., Gengler N., Bertrand J.K., Varona L., Estimation of dominance variance in purebred yorkshire swine, J. Anim. Sci. 76 (1998) 448–451.

[7] DeBoer I.J.M., Hoeschele I., Genetic evaluation methods for populations with dominance and inbreeding, Theor. Appl. Genet. 86 (1993) 245–258.
[8] Du F.X., Hoeschele I., Estimation of additive, dominance and epistatic variance components using finite locus models implemented with a single-site gibbs and a descent graph sampler, Genet. Res. 76 (2000) 187–198.
[9] Elston R.C., Stewart J., A general model for the genetic analysis of pedigree data, Hum. Hered. 21 (1971) 523–542.
[10] Falconer D.S., Mackay T.F.C., Introduction to quantitative genetics, Longman, Inc., New York, 4th edn., 1996.
[11] Fernandez S.A., Fernando R.L., Gulbrandtsen B., Totir L.R., Carriquiry A.L., Sampling genotypes in large pedigrees with loops, Genet. Sel. Evol. 33 (2001) 337–367.
[12] Fernando R.L., Theory for analysis of multi-breed data, in: Proceedings for the 7th Genetic Prediction Workshop, 1999, Kansas City, MO, USA, pp. 1–16.
[13] Fernando R.L., Gianola D., Optimal properties of the conditional mean as a selection criterion, Theor. Appl. Genet. 72 (1986) 822–825.
[14] Fernando R.L., Grossman M., Genetic evaluation in crossbred populations, in: Proc. Forty-Fifth Annu. Natl. Breeders Roundtable, Poult. Breeders Am. and US Poult. Egg Assoc., 1996, Tucker, GA, pp. 19–28.
[15] Fernando R.L., Stricker C., Elston R.C., An efficient algorithm to compute the posterior genotypic distribution for every member of a pedigree without loops, Theor. Appl. Genet. 87 (1993) 89–93.
[16] Fernando R.L., Stricker C., Elston R.C., The finite polygenic mixed model: An alternative formulation for the mixed model of inheritance, Theor. Appl. Genet. 88 (1994) 573–580.
[17] Gianola D., Fernando R.L., Bayesian methods in animal breeding, J. Anim. Sci. 63 (1986) 217–244.
[18] Gilks W.R., Richardson S., Spiegelhalter D.J., Introducing Markov chain Monte Carlo, in: Gilks W.R., Richardson S., Spiegelhalter D.J. (Eds.), Markov chain Monte Carlo in practice, 1996, 2–6 Boundry Row, London SE1 8HN, Chapman & Hall, pp. 1–16.
[19] Goddard M.E., Gene based models for genetic evaluation – an alternative to blup?, in: Proceedings of the 6th World Congress on Genetics Applied to Livestock Production, Armidale, 11–16 January 1998, Vol. 26, University of New England, Armidale, pp. 33–36.
[20] Heath S.C., Markov chain Monte Carlo segregation and linkage analysis for oligogenic models, Am. J. Hum. Genet. 61 (1997) 748–760.
[21] Janss L.L.G., Thompson R., van Arendonk J.A.M., Applications of Gibbs sampling for inference in a mixed major gene-polygenic inheritance model in animal populations, Theor. Appl. Genet. 91 (1995) 1137–1147.
[22] Jensen C.S., Kong A., Kjærulff U., Blocking Gibbs sampling in very large probabilistic expert systems, Int. J. Hum. Comp. Stud. 42 (1995) 647–666.
[23] Meuwissen T.H.E., Goddard M.E., The use of marker haplotypes in animal breeding schemes, Genet. Sel. Evol. 28 (1996) 161–176.
[24] Perez-Enciso M., Varona L., Rothschild M.F., Computation of identity by descent probabilities conditional on DNA markers via Monte Carlo Markov chain method, Genet. Sel. Evol. 32 (2000) 467–482.

[25] Perez-Enciso M., Fernando R.L., Bidanel J.P., Le Roy P., Quantitative trait locus analysis in crosses between outbred lines with dominance and inbreeding, Genetics 159 (2001) 413–422.
[26] Ramsey F.L., Schafer D.W., The statistical sleuth a course in methods of data analysis, Duxbury Press, 1st edn., 1997.
[27] Smith C., Improvement of metric traits through specific genetic loci, Anim. Prod. 9 (1967) 349–358.
[28] Stricker C., Fernando R.L., Some theoretical aspects of finite locus models, in: Proceedings of the 6th World Congress on Genetics Applied to Livestock Production, Armidale, 11–16 January 1998, Vol. 26, University of New England, Armidale, pp. 25–32.
[29] Thompson E.A., Heath S.C., Estimation of conditional multilocus gene identity among relatives, in: Statistics in Molecular Biology, IMS Lecture Notes – Monograph Series, Vol. 33, 1999, pp. 95–113.
[30] Totir L.R., Fernando R.L., Fernandez S.A., The effect of the number of loci on genetic evaluations in finite locus models, J. Anim. Sci. 79 (Suppl. 1) (2001) 191.
[31] Totir L.R., Genetic evaluation with finite locus models, Ph.D. thesis, Iowa State University, 2002.
[32] van Arendonk J.A.M., Smith C., Kennedy B.W., Method to estimate genotype probabilities at individual loci farm livestock, Theor. Appl. Genet. 78 (1989) 735–740.
[33] Wang T., Fernando R.L., Stricker C., Elston R.C., An approximation to the likelihood for a pedigree with loops, Theor. Appl. Genet. 93 (1996) 1299–1309.