

A LOOK AT COMPUTATIONS FOR MULTIVARIATE SINGLE-STEP GENOMIC EVALUATIONS FITTING THE ‘HYBRID MODEL’

Karin Meyer

Animal Genetics and Breeding Unit*, University of New England, Armidale NSW 2351, Australia

SUMMARY

Computational requirements for single step genomic evaluation fitting a hybrid between breeding value and marker effects models are examined for a simulated example. It is demonstrated that such a model can accommodate large numbers of genotyped animals – readily allowing exploitation of large in-core memory and parallel processing capabilities available with modern hardware – and that a principal component parameterization for multivariate analyses of numerous traits is advantageous.

INTRODUCTION

Genomic evaluation jointly considering genotyped and non-genotyped animals in a so-called single-step (SS) analysis has become routine procedure for many genetic evaluation schemes. Most implementations invoke a formulation which ‘simply’ replaces the pedigree based relationship matrix in the standard ‘breeding value’ (BV) model with its counterpart incorporating genomic information; see Legarra *et al.* (2014) for a review. Recently, Fernando *et al.* (2014, 2016) proposed an alternative which does not require construction or inversion of a genomic relationship matrix: the ‘hybrid model’ (HM) combines a BV model for non-genotyped animals with a ‘marker effects’ model for genotyped individuals to represent additive genetic effects. Describing strategies for efficient computations, the authors emphasized the scope of the HM to exploit the parallel processing capacities of modern hardware. This paper presents a first look at computational demands of multivariate genomic evaluation under the HM, including an evaluation of a parameterisation to principal components.

THE HYBRID MODEL

Consider records for q traits and let subscripts ‘1’ and ‘2’ denote terms pertaining to n_1 non-genotyped and n_2 genotyped individuals, respectively. Let $\mathbf{I}_q$ denote an identity matrix of size q . Ordering genetic effects by individuals or markers within traits, the multivariate HM model is

$$\begin{aligned} \mathbf{y}_1 &= \mathbf{X}_1 \mathbf{b} + \mathbf{Z}_1 \mathbf{u}_1 + \mathbf{e}_1 = \mathbf{X}_1 \mathbf{b} + \mathbf{Z}_1 [(\mathbf{I}_q \otimes \mathbf{M}_1) \boldsymbol{\alpha} + \boldsymbol{\epsilon}] + \mathbf{e}_1 \\ \mathbf{y}_2 &= \mathbf{X}_2 \mathbf{b} + \mathbf{Z}_2 (\mathbf{I}_q \otimes \mathbf{M}_2) \boldsymbol{\alpha} + \mathbf{e}_2 \end{aligned} \quad (1)$$

with $\mathbf{y}_i$, $\mathbf{b}$, $\mathbf{u}_i$, $\boldsymbol{\alpha}$ and $\mathbf{e}_i$ the vectors of records, fixed effects, breeding values, marker effects and residuals, $\mathbf{X}_i$ and $\mathbf{Z}_i$ the corresponding incidence matrices, and $\mathbf{M}_i$ the matrices of marker counts, appropriately centered and scaled. Marker counts for non-genotyped individuals need to be imputed. This can be done by regression using pedigree information, $\mathbf{M}_1 = \mathbf{A}_{12} \mathbf{A}_{22}^{-1} \mathbf{M}_2$, with $\mathbf{A}_{ij}$ the ij -th submatrix of the numerator relationship matrix and $\boldsymbol{\epsilon}$ accounting for imputation errors (Fernando *et al.* 2014). As formulated, the HM implies that breeding values for genotyped individuals are explained entirely by the markers fitted, but (1) is readily expanded to include additional polygenic effects if this does not hold or any other, additional random effects. Assuming $\text{Var}(\mathbf{u}_1) = \boldsymbol{\Sigma}_G \otimes \mathbf{A}_{11}$, $\text{Var}(\boldsymbol{\epsilon}) \approx \boldsymbol{\Sigma}_G \otimes (\mathbf{A}^{11})^{-1}$ and $\text{Var}(\boldsymbol{\alpha}) = \boldsymbol{\Sigma}_\alpha \otimes \mathbf{D}$, mixed model equations (MME) pertaining to (1) are

$$\begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'_1\mathbf{R}_1^{-1}\mathbf{Z}_1 & \mathbf{X}'_2\mathbf{R}_2^{-1}(\mathbf{I}_q \otimes \mathbf{M}_2) \\ \mathbf{Z}'_1\mathbf{R}_1^{-1}\mathbf{X}_1 & \mathbf{Z}'_1\mathbf{R}_1^{-1}\mathbf{Z}_1 + \boldsymbol{\Sigma}_G^{-1} \otimes \mathbf{A}^{11} & \boldsymbol{\Sigma}_G^{-1} \otimes \mathbf{A}^{12}\mathbf{M}_2 \\ (\mathbf{I}_q \otimes \mathbf{M}_2')\mathbf{Z}'_2\mathbf{R}_2^{-1}\mathbf{X}_2 & \boldsymbol{\Sigma}_G^{-1} \otimes \mathbf{M}_2'\mathbf{A}^{21} & (\mathbf{I}_q \otimes \mathbf{M}_2')\mathbf{Z}'_2\mathbf{R}_2^{-1}\mathbf{Z}_2(\mathbf{I}_q \otimes \mathbf{M}_2) \\ & & + \boldsymbol{\Sigma}_G^{-1} \otimes \mathbf{M}_1'\mathbf{A}^{11}\mathbf{M}_1 + \boldsymbol{\Sigma}_\alpha^{-1} \otimes \mathbf{D}^{-1} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{b}} \\ \hat{\mathbf{u}}_1 \\ \hat{\boldsymbol{\alpha}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{y} \\ \mathbf{Z}'_1\mathbf{R}_1^{-1}\mathbf{y}_1 \\ (\mathbf{I}_q \otimes \mathbf{M}_2')\mathbf{Z}'_2\mathbf{R}_2^{-1}\mathbf{y}_2 \end{bmatrix} \quad (2)$$

*AGBU is a joint venture of NSW Department of Primary Industries and the University of New England

PC formulation. A parameterisation to principal components (PC) is obtained by replacing $\mathbf{Z}_i$ with $\mathbf{Z}_i^* = \mathbf{Z}_i(\mathbf{Q} \otimes \mathbf{I})$, $\mathbf{u}_1$ with $\mathbf{u}_1^* = (\mathbf{Q}^{-1} \otimes \mathbf{I})\mathbf{u}_1$ and α with $\alpha^* = (\mathbf{Q}^{-1} \otimes \mathbf{I})\alpha$. A suitable choice is $\mathbf{Q} = \mathbf{E}\mathbf{A}^{1/2}\mathbf{T}$, the ‘factor matrix’ obtained from the eigen-decomposition of $\Sigma_G = \mathbf{E}\mathbf{A}\mathbf{E}'$ with orthogonal rotation $\mathbf{T}$ to lower triangular form (Meyer *et al.* 2015). Truncating $\mathbf{Q}$ to $r \leq q$ columns, this replaces Σ_G^{-1} in (2) with $\mathbf{I}_r$ and Σ_α^{-1} with $\mathbf{Q}'\Sigma_\alpha^{-1}\mathbf{Q}$, where $\mathbf{I}_r$ denotes an identity matrix of size r .

Computational strategies. Consider the iterative solution of (2) using a preconditioned conjugate gradient (PCG) algorithm. This requires the product of the coefficient matrix in the MME, $\mathbf{C}$, with a vector, $\mathbf{r}$, in each iterate, $\mathbf{C}\mathbf{r} = \mathbf{q}$. Generally, $\mathbf{C}$ is too large to be stored in core. Partition $\mathbf{C}$, $\mathbf{r}$ and $\mathbf{q}$ according to the three types of effects fitted, dropping the subscript ‘1’ from $\mathbf{u}_1$ in the following

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{bb} & \mathbf{C}_{bu} & \mathbf{C}_{bu\alpha} \\ \mathbf{C}_{ub} & \mathbf{C}_{uu} & \mathbf{C}_{u\alpha} \\ \mathbf{C}_{\alpha b} & \mathbf{C}_{\alpha u} & \mathbf{C}_{\alpha\alpha} \end{bmatrix} \quad \mathbf{r} = \begin{bmatrix} \mathbf{r}_b \\ \mathbf{r}_u \\ \mathbf{r}_\alpha \end{bmatrix} \quad \text{and} \quad \mathbf{q} = \begin{bmatrix} \mathbf{q}_b \\ \mathbf{q}_u \\ \mathbf{q}_\alpha \end{bmatrix} \quad (3)$$

Submatrices of $\mathbf{C}$ corresponding to $\mathbf{b}$ and $\mathbf{u}$ are the same as in the BV model, i.e. the respective parts of $\mathbf{C}\mathbf{r}$ can be evaluated using sparse matrix multiplication or, for large problems, standard ‘iteration on data’ techniques. Fernando *et al.* (2016) considered a scenario where the number of markers (m) is relatively small – in the tens rather than hundreds of thousands – so that both $\mathbf{C}_{\alpha\alpha}$ and $\mathbf{M}_2$ of size $n_2 \times m$ could be stored in core. For more than a few traits, however, the former can be too large and even evaluating its distinct $q(q+1)/2$ submatrices (for pairs of traits) once and loading them from out-of-core storage in each PCG iterate may be impracticable. Yet, $\mathbf{M}_1\mathbf{A}^{11}\mathbf{M}_1$ of size $m \times m$ may be held in core. Fernando *et al.* (2016) described how to impute columns of $\mathbf{M}_1$ for individual markers and how to obtain this product efficiently without the need to store $\mathbf{M}_1$. The authors further emphasized evaluation of partial products, required in solving the MME, in steps. For instance, $\mathbf{C}_{uu}\mathbf{r}_\alpha = (\Sigma_G^{-1} \otimes \mathbf{A}^{12}\mathbf{M}_2)\mathbf{r}_\alpha$ can be separated into dense matrix $\times$ sub-vector products for trait i , $\mathbf{M}_2\mathbf{r}_{\alpha,i} = \mathbf{t}_i$, followed by sparse products $\mathbf{A}^{12}\mathbf{t}_i$, and finally pre-multiplication of the resulting vector for all traits with $\Sigma_G^{-1} \otimes \mathbf{I}_{n_1}$. This eliminates the need to store the large, dense matrix $\mathbf{M}_2\mathbf{A}^{21}$ of size $m \times n_1$. Moreover, evaluating a product of form $\mathbf{S}\mathbf{W}\mathbf{t}$ (for variable $\mathbf{t}$) in steps as $\mathbf{S}(\mathbf{W}\mathbf{t})$ can be more efficient than $(\mathbf{S}\mathbf{W})\mathbf{t}$ (Strandén and Lidauer 1999). Similar decompositions can be employed for the remaining products, exploiting that $(\mathbf{I}_q \otimes \mathbf{M}_2)\mathbf{r}_\alpha$ occurs multiple times and that multiplication with $\mathbf{I}_q \otimes \mathbf{M}_2'$ can be applied to the sum of partial vectors, i.e both computationally intensive products are only needed once per PCG iterate. Detailed steps are summarised in Figure 1.

APPLICATION

To evaluate performance of the HM, data for $q = 16$ traits recorded on 1.5 million animals in 3 generations with $m = 20000$ markers were simulated using AlphaSim (Faux *et al.* 2016). Genetic and environmental correlations among traits were assumed to be 0.6 and 0.3 throughout, while heritabilities for odd and even numbered traits were set to 0.5 and 0.2, respectively. Analyses fitted 37,500 randomly assigned fixed contemporary groups and restricted the marker information utilised to randomly selected animals in generation 3, ranging from $n_2 = 0$ to 500000.

Marker counts were centered using frequencies estimated from the data. $\mathbf{M}_1$ was imputed by solving $\mathbf{A}^{11}\mathbf{M}_1 = -\mathbf{A}^{12}\mathbf{M}_2$ (Fernando *et al.* 2016) using sparse matrix factorisation of $\mathbf{A}^{11}$ after reordering to minimise fill-in and triangular solves for blocks of $s = 40, 100$ or 200 markers at a time. Iterative solutions of the MME were obtained using a PCG algorithm with diagonal preconditioner and convergence criterion of 10^{-7} , setting genetic and residual covariances to values reported by the simulation program, assuming $\Sigma_\alpha = \frac{1}{m}\Sigma_G$ and $\mathbf{D} = \mathbf{I}_m$. Analyses used the standard multivariate parameterisation shown above (MV16) or parameterised to r principal component (PCr) for $r = 16, 14$ and 12. Computations were carried out under Linux on a shared machine with 512GB of RAM and

1	$\begin{bmatrix} \mathbf{q}_b \\ \mathbf{q}_u \end{bmatrix} := \begin{bmatrix} \mathbf{C}_{bb} & \mathbf{C}_{bu} \\ \mathbf{C}_{ub} & \mathbf{Z}'_1 \mathbf{R}_1^{-1} \mathbf{Z}_1 \end{bmatrix} \begin{bmatrix} \mathbf{r}_b \\ \mathbf{r}_u \end{bmatrix}$	6	$\mathbf{q}_u := \mathbf{q}_u + (\Sigma_G^{-1} \otimes \mathbf{I}_{n_1}) \mathbf{w}$	11	$\mathbf{t} := \mathbf{Z}'_2 \mathbf{R}_2^{-1} \mathbf{Z}_2 \mathbf{t}$
2	$\mathbf{t} := (\mathbf{I}_q \otimes \mathbf{M}_2) \mathbf{r}_\alpha$	7	$\mathbf{w} := (\mathbf{I}_q \otimes \mathbf{M}'_1 \mathbf{A}^{11} \mathbf{M}_1) \mathbf{r}_\alpha$	12	$\mathbf{t} := \mathbf{t} + \mathbf{Z}'_2 \mathbf{R}_2^{-1} \mathbf{X}_2 \mathbf{r}_b$
3	$\mathbf{q}_b := \mathbf{q}_b + \mathbf{X}'_2 \mathbf{R}_2^{-1} \mathbf{Z}_2 \mathbf{t}$	8	$\mathbf{q}_\alpha := (\Sigma_G^{-1} \otimes \mathbf{I}_m) \mathbf{w}$	13	$\mathbf{w} := (\mathbf{I}_q \otimes \mathbf{A}^{21}) \mathbf{r}_u$
4	$\mathbf{w} := (\mathbf{I}_q \otimes \mathbf{A}^{12}) \mathbf{t}$	9	$\mathbf{w} := (\mathbf{I}_q \otimes \mathbf{D}^{-1}) \mathbf{r}_\alpha$	14	$\mathbf{t} := \mathbf{t} + (\Sigma_G^{-1} \otimes \mathbf{I}_{n_2}) \mathbf{w}$
5	$\mathbf{w} := \mathbf{w} + (\mathbf{I}_q \otimes \mathbf{A}^{11}) \mathbf{r}_u$	10	$\mathbf{q}_\alpha := \mathbf{q}_\alpha + (\Sigma_\alpha^{-1} \otimes \mathbf{I}_m) \mathbf{w}$	15	$\mathbf{q}_\alpha := \mathbf{q}_\alpha + (\mathbf{I}_q \otimes \mathbf{M}'_2) \mathbf{t}$

Figure 1. Steps to compute of $\mathbf{C}\mathbf{r} = \mathbf{q}$ without storing sub-matrices $\mathbf{C}_{\alpha,\alpha}$, $\mathbf{C}_{\alpha\cdot}$ or $\mathbf{C}_{\cdot\alpha}$ in core.

28 Intel Xeon CPU E5-2697 cores, rated at 2.6Ghz, with a cache size of 35MB. Calculations were performed exploiting BLAS and sparse BLAS routines and the parallel direct sparse solver PARDISO to impute $\mathbf{M}_1$, all loaded from the multi-threaded Intel Math Kernel Library 11.3.2. In addition, OpenMP directives were employed to parallelise selected operations, using up to 28 threads. Sparse matrices $\mathbf{C}_{bb}$, $\mathbf{C}_{bu}$, $\mathbf{C}_{ub}$, $\mathbf{Z}'_1 \mathbf{R}_1^{-1} \mathbf{Z}_1$, $\mathbf{X}'_2 \mathbf{R}_2^{-1} \mathbf{Z}_2$, $\mathbf{Z}'_2 \mathbf{R}_2^{-1} \mathbf{Z}_2$, $\mathbf{A}^{11}$, $\mathbf{A}^{12}$ and $\mathbf{A}^{21}$ were held in core, using compressed matrix storage. Whilst $\mathbf{C}_{bu} = \mathbf{C}'_{ub}$ and $\mathbf{A}^{12} = \mathbf{A}^{21'}$, holding the additional transposed copies allowed better use of BLAS routines for parallel computations at little increase in RAM required.

RESULTS

Computational requirements to determine $\mathbf{M}'_1 \mathbf{A}^{11} \mathbf{M}_1$ and to solve the MME for increasing numbers of genotyped animals are summarised in Table 1, together with selected characteristics of the MME. All times shown are elapsed (wall) times for the specific task, excluding set-up steps.

Most memory (RAM) used was for in-core storage of $\mathbf{M}_2$, of size $n_2 \times m$, while holding $\mathbf{M}'_1 \mathbf{A}^{11} \mathbf{M}_1$ for $m = 20000$ in core required just under 3GB (full-stored). Imputation of $\mathbf{M}_1$ together with calculation of $\mathbf{M}'_1 \mathbf{A}^{11} \mathbf{M}_1$ required less than half an hour, with some advantage for larger numbers of markers processed at once and some increase in time required with growing n_2 . In comparison, building and inverting the genomic relationship matrix for the same markers required 6, 43, 149 and 381 minutes for $n_2 = 50, 100, 150$ and 200K, respectively.

Not surprisingly, including genomic information in genetic evaluation increases computational demands to solve the MME by orders of magnitude, the more so the greater the proportion of genotyped individuals. Not only are there substantially more operations per iterate, but for the HM with many non-zero off-diagonal elements in $\mathbf{C}_{\alpha\alpha}$ solutions converged slowly resulting in many more PCG iterates to be performed. Employing a simple, diagonal preconditioner in the PCG algorithm, parameterising to genetic principal components reduces the number of iterates and thus time required substantially. This is due to ‘de-correlating’ genetic effects for different traits and thus can be less effective when genetic correlations between traits are weak or when more sophisticated conditioning schemes are applied (Meyer 2016). If the number of PCs fitted can be reduced, both RAM and computing time required are decreased further, mainly due to a reduction in the number of equations and therefore the number of operations per iterate. In comparison, a corresponding SS analysis fitting a BV model for $n_2 = 50\text{K}$ and MV16 required only 290 iterates and 10 minutes for the solution phase.

DISCUSSION

In practice, genetic evaluation models are more complex than considered here and additional random effects – especially genetic groups – are likely to increase the resources required for HM analyses markedly. Nevertheless, results illustrate that multivariate evaluation under the HM is feasible for numerous traits and many thousands of genotyped animals, especially if aided by a parameterisation to genetic principal components. Computational demands are proportional to the number of markers considered, i.e. may necessitate research efforts to identify appropriate subsets.

Table 1. Computational requirements for genomic evaluation via the hybrid model

		No. of genotyped animals, n_2 (K)								
		0	25	50	75	100	150	200	300	500
Genotyped animals (%)		0	1.7	3.3	5.0	6.7	10.0	13.3	20.0	33.3
No. of equations (M)		24.6	24.5	24.1	23.7	23.3	22.5	21.7	20.1	16.9
RAM to store $\mathbf{M}_2$ (GB)		–	3.7	7.4	11.2	14.9	22.4	29.8	44.7	74.5
Imputation of $\mathbf{M}_1$ and calculation of $\mathbf{M}_1' \mathbf{A}^{-1} \mathbf{M}_1$										
RAM (GB)	40 ^a	–	26	30	33	37	45	52	67	96
	100	–	29	33	36	40	47	54	69	98
	200	–	33	37	40	44	51	58	73	102
Time (min)	40	–	23	24	24	24	25	26	27	29
	100	–	17	17	17	19	20	21	21	21
	200	–	15	15	15	16	16	16	19	19
Solution of mixed model equations										
RAM (GB)	MV16	35	42	45	48	52	59	66	81	108
	PC16	27	34	37	41	45	52	59	74	102
	PC14	25	32	36	39	43	50	58	73	101
	PC12	23	30	34	38	41	49	56	71	99
No. of iterates	MV16	350	1249	1395	1449	1457	1534	1707	1997	2302
	PC16	167	597	623	638	629	654	668	668	678
	PC14	165	584	624	612	624	653	654	659	680
	PC12	163	572	618	624	635	646	654	653	671
Time (min)	MV16	4	23	33	42	51	66	84	129	237
	PC16	2	15	18	22	29	34	44	68	100
	PC14	2	10	14	18	21	28	35	47	81
	PC12	1	9	13	16	19	24	30	41	68

^aBlock size for imputation of markers

For beef cattle, Saatchi and Garrick (2016) proposed a reduced panel comprising about 2,300 markers and reported predictive performance of more than 80% of that for a full 50K panel. For our example, analyses fitting a BV model converged more quickly, presumably in part due to inclusion of non-important markers in the HM analysis. The HM can be implemented exploiting efficient, off-the shelf linear algebra routines. It appears best suited to analyses with large numbers of genotyped animals where a sparse approximation of the inverse of the genomic relationship matrix is not desirable.

ACKNOWLEDGEMENTS

Work was supported by Meat and Livestock Australia grant L.GEN.1704.

REFERENCES

- Faux A.M., Gorjanc G., Gaynor R.C., Battagin M., Edwards S.M., Wilson D.L., Hearne S.J., Gonen S. and Hickey J.M. (2016) *Plant Genome* **9**:1.
- Fernando R.L., Cheng H., Golden B.L. and Garrick D.J. (2016) *Genet. Sel. Evol.* **48**:96.
- Fernando R.L., Dekkers J.C.M. and Garrick D.J. (2014) *Genet. Sel. Evol.* **46**:50.
- Legarra A., Christensen O.F., Aguilar I. and Misztal I. (2014) *Livest. Sci.* **166**:54.
- Meyer K. (2016) *J. Anim. Sci.* **94**:4530.
- Meyer K., Swan A. and Tier B. (2015) *J. Anim. Sci.* **93**:4624.
- Saatchi M. and Garrick D.J. (2016) In *Plant and Animal Genome XXIV Conference*. San Diego, CA, January 9–13, 2016. Abstr.
- Strandén I. and Lidauer M. (1999) *J. Dairy Sci.* **82**:2779.