Proceedings of the World Congress on Genetics Applied to Livestock Production, 11, 25

BLUP without (inverse) relationship matrix

E. Groeneveld[(1)] *and A. Neumaier*[(2)]

[(1)] *Institute of Farm Animal Genetics, Friedrich-Loeffler-Institut, Höltystr. 10 D-31535 Neustadt a. Rbge., Germany*
email: eildert.groeneveld@gmx.de

[(2)] *Faculty of Mathematics, University of Vienna, Oskar-Morgenstern-Platz 1, A-1090 Vienna, Austria*

Summary

Mixed model methodology provides the machinery for genetic evaluation in animal breeding. In fact they are the basis of all modern methods for obtaining estimates of their unknowns. One central component of Henderson's Mixed Model Equations (MME) is the numerator relationship matrix (NRM) and, more precisely, its inverse. Only after Henderson found a way to directly set up the inverse, the animal model became computationally feasible.

We show that there is a way to totally avoid the modelling of the NRM or its inverse, yet arriving at exactly the same BLUPs and BLUEs. This reduces the program complexity and the computational complexity.

In practical modelling, it is useful to split the general model into blocks of uncorrelated model equations, here called element equations. Each such element equation consists of a small number of (possibly only one) correlated equations linked by their covariance matrix, which are block diagonal and thus trivial to invert; multiple trait models being an example. In their NRM-free form discussed in the present paper, standard animal models have three types of model equations: phenotype based, trivial for simple random effects, and pedigree model equations. The coefficients derived for the pedigree model equation are, indeed, the same as those given by Westell et al. (1987) without having to deal with the relationship matrix separately.

BLUPs can be computed solely on the basis of these element equations: once the model equations have been set up in terms of coefficients and mixed model addresses and their covariance matrix association, further processing is oblivious to the statistical model and becomes a very simple sweep. Only a few lines of code are required for either setting up the complete MME in sparse format for a direct solution and possibly computing its inverse, or in conjugate gradients to iteratively solve for BLUEs and BLUPs.

A small numerical example is given to describe the process.

The model equations are well suited for coarse-grained parallelization. As implemented in PEST2, they scale well with little computing overhead, therefore making good use of multi core computers.

An outlook is given for the use of the method for the handling of genomic (SNP) data in genetic evaluation. Also here, explicit treatment of the NRM and GRM is not required, leading to a joint uniform one step evaluation with pedigree and genomic data with no additional assumptions except the model and covariances.

Introduction

Mixed Models provide the machinery for genetic evaluation in animal breeding. One central component is the numerator relationship matrix (NRM) and more precisely its inverse. Only after Henderson found a way to directly set up the inverse, the animal model became computationally feasible. There is, however, a way to totally avoid the treatment of the NRM, yet arriving at exactly the same BLUPs. Avoiding the NRM implies not having to deal with its inverse, something that becomes particularly interesting in the context of genomic evaluation using the genomic relationship matrix, which does indeed up to now require a computed inverse. This contribution presents the framework of model equations as an alternative approach for computing solutions to mixed models without having to explicitly consider the NRM or its inverse.

Linear stochastic models

Many applications (including those to animal breeding) are based on the **general linear stochastic model**

$$y = X\beta + Zu + \eta, \quad \mathrm{cov}(u) = G, \quad \mathrm{cov}(\eta) = D, \tag{1}$$

with **fixed effects** β, **random effects** u and noise η. Here $\mathrm{cov}(u)$ denotes the covariance matrix of a random vector u with zero mean. Usually, D is block diagonal, with many identical blocks. In many applications, G is also block diagonal. In animal breeding, G is (in the absence of nongenetic random effects) the NRM, which is large and typically dense. However, in this case the inverse G^{-1} is sparse and easily computable from the pedigree of the animals involved.

Mixed model equations

By combining the two noise terms, the model is seen to be equivalent to the simple model $y = X\beta + \eta'$, where η' is a random vector with zero mean and (mixed model) covariance matrix $V = ZGZ^T + D$. Usually, V is a huge and dense matrix, leading to hardly manageable normal equations involving the inverse of V. However, Henderson (1950) showed that the normal equations are equivalent to the **mixed model equations (MME)**

$$\begin{pmatrix} X^TD^{-1}X & X^TD^{-1}Z \\ Z^TD^{-1}X & Z^TD^{-1}Z + G^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ u \end{pmatrix} = \begin{pmatrix} X^TD^{-1}y \\ Z^TD^{-1}y \end{pmatrix}. \tag{2}$$

This formulation avoids the inverse of the mixed model covariance matrix V and involves instead the well-behaved inverse relationship matrix G^{-1}. It is the basis of all modern methods for obtaining estimates of u and β in (1).

Fellner (1986) observed that Henderson's mixed model equations are the normal equations of an augmented model of the simple form

$$Ax = b + \mathrm{noise}(C), \tag{3}$$

where

$$x = \begin{pmatrix} \beta \\ u \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} -\eta \\ u \end{pmatrix},$$

$$A = \begin{pmatrix} X & Z \\ 0 & I \end{pmatrix}, \quad b = \begin{pmatrix} y \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} D & 0 \\ 0 & G \end{pmatrix},$$

and $\mathrm{noise}(C)$ stands for a vector of random noise with mean zero and covariance matrix C. Note that C^{-1} is sparse whenever D and G are block diagonal, but also when only D is block diagonal and G^{-1} is sparse.

Thus, without loss in generality, we may base our algorithms on the simple model (3), with a very sparse inverse covariance matrix C^{-1}. This automatically produces the formulas that previously had to be derived in a less transparent way by means of the W transformation; cf. (Hemmerle and Hartley, 1973; Corbeil and Searle, 1976; Wolfinger et al., 1994; Fraley and Burns, 1995).

The **normal equations** for the model (3) have the form

$$Bx = r, \tag{4}$$

where

$$B = A^T C^{-1} A, \quad r = A^T C^{-1} b.$$

Here A^T denotes the transposed matrix of A. Note that only C^{-1} figures(C being block diagonal and can, thus be easily inverted); so the normal equations are efficiently computable; their coefficient matrix is typically still very sparse since both A and C^{-1} are very sparse. By solving the normal equations (4), we obtain the best linear unbiased prediction (BLUP)

$$\widehat{x} = B^{-1} r = B^{-1} A^T C^{-1} b \tag{5}$$

for the vector x, and the noise $\varepsilon = Ax - b$ is estimated by the residual

$$\widehat{\varepsilon} = A\widehat{x} - b.$$

For the practical modeling of linear stochastic systems, it is useful to split a model (3) into blocks of uncorrelated model equations which we call **element equations**. Each element equations consists of a few correlated equations only (possibly only one); these usually fall into several types, distinguished by their covariance matrices. The model equation for an element ν of type γ has the form

$$A_\nu x = b_\nu + \mathrm{noise}(C_\gamma). \tag{6}$$

Here A_ν is the coefficient matrix of the block of equations for element number ν. Generally, A_ν is very sparse with few rows and many columns, most of them zero, since only a small subset of the variables occurs explicitly in the νth element.

Each model equation has only *one* noise term. Correlated noise must be put into one element. All elements of the same type are assumed to have statistically independent noise vectors, realizations of (not necessarily Gaussian) distributions with zero mean and the same covariance matrix. Thus the various elements are assigned to the types according to the covariance matrices of their noise vectors.

Treating all measurements as belonging to one type γ with only one covariance matrix may generate large numbers of nonzero entries in the normal equations matrix, unduly increasing both CPU time and memory requirements. When measurements are taken in different environments the errors between these groups are therefore assumed uncorrelated, leading to a very sparse, block diagonal inverse covariance matrix and hence to sparse normal equations.

Computing BLUEs and BLUPs

For elements numbered by $\nu = 1, ..., N$, the full matrix formulation of the model (6) is the model (3) with

$$A = \begin{pmatrix} A_1 \\ \vdots \\ A_N \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_N \end{pmatrix}, \quad C = \begin{pmatrix} C_{\gamma(1)} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & C_{\gamma(N)} \end{pmatrix},$$

where $\gamma(\nu)$ denotes the type of element ν.

In case of some missing data we may proceed very similar as in Neumaier and Groeneveld (1998), but with a slightly improved notation. The incomplete model obtained by deleting from the full element formulation (6) all equations containing missing data has the **incomplete element equations**

$$PA'_\nu x = P_\pi b'_\nu + \text{noise}(C'_\nu), \tag{7}$$

where

$$A'_\nu := P_\pi A_\nu, \quad b'_\nu = P_\pi b_\nu, \quad C'_\nu := P_\pi C_{\gamma(\nu)} P_\pi^T, \tag{8}$$

and the P_π are projection operators that remove the rows corresponding to the missing data. The missing data patterns are indexed by π and may correspond to data missing systematically, or to data missing at random; in the latter case they depend not on the generic model but on the specific data set. The resulting incomplete model has the form (3) with

$$A = \begin{pmatrix} A'_1 \\ \vdots \\ A'_N \end{pmatrix}, \quad b = \begin{pmatrix} b'_1 \\ \vdots \\ b'_N \end{pmatrix}, \quad C = \begin{pmatrix} C'_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & C'_N \end{pmatrix}.$$

In conjugate gradients, the diagonal $D = \operatorname{Diag} B$ of B can be used as a preconditioner. In multiple trait models the diagonal blocks reflecting the traits involved can be used for faster convergence at the expense of slightly higher computing cost of each.

Using the precomputed preconditioner D and the matrix-vector product routine (Bp) with M_π being the inverse of the corresponding C block:

```
% compute matrix vector product q = Bp
q = 0;
do for all types γ
    do for all elements ν of type γ
        get the missing data pattern π of element ν
        gather A' = P_π A_ν and p' = p_ν;
        q' = A'^T (M_π (A' p')); q = q ⊕ q';
    end
end
% return q
```

we apply the preconditioned conjugate gradient method for solving the linear system $Bx = r$ with a symmetric, positive definite coefficient matrix B as follows; cf. Algorithm 10.3.1 and (10.2.6) in Golub and van Loan (2012).

Algorithm: Conjugate gradient method (CG)

Purpose: Solves a positive definite linear system $Bx = r$
Input: r (right hand side), ε (requested relative accuracy),
evaluator for $q = Bp$, solver for $Ds = r$ (preconditioner)
Requirements: B, D symmetric and positive semidefinite,
D nonsingular

$p = 0;\ x = 0;\ \omega = \infty;\ \Delta = 0;$
while 1,
  solve $Ds = r$ for s;
  $\omega_{\text{old}} = \omega;\ \omega = r^T s;$
  if $\omega \leq 0$, return; end;
  $\beta = \omega/\omega_{\text{old}};\ p = \beta p + s;\ q = Bp;\ \alpha = p^T r/p^T q;$
  $x = x + \alpha p;\ \delta = \alpha^2 \omega;$
  if $\delta \leq \Delta$, return; end;
  $r = r - \alpha q;\ \Delta = \max(\varepsilon^2 \delta, \Delta);$
end;

The effort per iteration consists of one matrix-vector multiplication $q = Bp$, one application of the preconditioner (solve $Ds = r$ for s), and $12n + O(1)$ other operations, where n is the number of variables.

Element formulation

In the estimation problems from animal breeding, the vector x splits into small vectors β_k of (in our present implementation constant) size n_{trait} called **effects**. Part of the right-hand side b contains measured data vectors y_ν; the remaining part consists of zeros. Each index ν corresponds to some animal; the n_{rec} **data records** y_ν^T are the rows of an $n_{rec} \times n_{trait}$ measurement matrix.

Typical element types – leading to a set of independent model equations – are as follows:

(i) **Measurement elements**: For those animals for which measurements are available, the measurement vectors $y_\nu \in \mathbb{R}^{n_{trait}}$ are explained in terms of a linear combination of effects $\beta_i \in \mathbb{R}^{n_{trait}}$,

$$y_\nu \approx \sum_{l=1}^{n_{eff}} \mu_{\nu l} \beta_{i_{\nu l}},$$

which leads to element equations

$$\sum_{l=1}^{n_{eff}} \mu_{\nu l} \beta_{i_{\nu l}} = y_\nu + \text{noise}(C_M).$$

Here the $i_{\nu l}$ form an $n_{rec} \times n_{eff}$ index matrix and the $\mu_{\nu l}$ form an $n_{rec} \times n_{eff}$ coefficient matrix.

(ii) **Pedigree elements**: For some animals, identified by the index T of their additive genetic effect β_T, parents may be known, with corresponding indices S (father) and D (mother). Their genetic dependence is modeled by a relation

$$\beta_{T(\nu)} \approx \frac{1}{2}\beta_{S(\nu)} + \frac{1}{2}\beta_{D(\nu)}$$

This leads to element equations

$$\beta_{T(\nu)} - \frac{1}{2}\beta_{S(\nu)} - \frac{1}{2}\beta_{D(\nu)} = 0 + \text{noise}(C_P).$$

The indices or addresses in the MME are stored in **pedigree records** that contain a column of animal indices $T(\nu)$ and two further columns for their parents $(S(\nu), D(\nu))$. However, there will always be animals for which only one or no parent is known. Thus the actual element equations are slightly more complicated as shown in the section on missing parents below.

(iii) **Random effect elements**: Certain effects $\beta_{R(\gamma)}$, are considered as random effects by including trivial model equations

$$\beta_{R(\gamma)} = 0 + \text{noise}(C_\gamma).$$

As part of the model (6), these trivial elements automatically produce the traditional mixed model equations, as explained above.

The inverse relationship matrix

To relate the form of the model to the mixed model (1) we assume for simplicity that there are no other random elements. Then we may write the measurement element equations in matrix form as

$$X\beta + Zu = y + \text{noise}(D), \quad D = C_M,$$

where β contains the fixed effects and u the additive genetic effects in the measurement equations, and the pedigree element equations as

$$Pu = \text{noise}(C_P). \tag{9}$$

This leads to the model (3) with

$$A = \begin{pmatrix} X & Z \\ 0 & P \end{pmatrix}, \quad b = \begin{pmatrix} y \\ 0 \end{pmatrix}, \quad x = \begin{pmatrix} \beta \\ u \end{pmatrix},$$

and $C = \text{Diag}(D, C_P)$. In this notation, the normal equations are $Bx = r$ with

$$B = \begin{pmatrix} B_{MM} & B_{MP} \\ B_{PM} & B_{PP} \end{pmatrix}, \quad r = \begin{pmatrix} X^T D^{-1} y \\ Z^T D^{-1} y \\ 0 \end{pmatrix}$$

with

$$B_{MM} = X^T D^{-1} X, \quad B_{MP} = X^T D^{-1} Z = B_{PM}^T,$$
$$B_{PP} = Z^T D^{-1} Z + P^T C_P^{-1} P,$$

The normal equations become

$$B_{MM}\beta + B_{MP}u = X^T D^{-1} y, \tag{10}$$

$$B_{PM}\beta + B_{PP}u = Z^T D^{-1} y, \tag{11}$$

Comparing (10) and (11) with (2) we see that if we choose

$$G^{-1} = P^T C_P^{-1} P, \tag{12}$$

where $P^T C_P^{-1} P$ is, up to a constant factor, the pedigree-based inverse relationship matrix, we recover Henderson's mixed model equation (2). Note that Fellner's observation mentioned above is the special case where in place of the pedigree equations (9) we have $u = \text{noise}(G)$, corresponding to $P = I$ and $C_P = G$, consistent with (12). Thus, with our way of representing the mixed model, the inverse relationship matrix is automatically and implicitly represented.

Model equations for missing parents

In this section we derive the model equations for breeding values with complete, partial or non existing pedigree. For generality, we assume that animals of unknown ancestry are classified into genetic groups; the case of a single genetic group covers the situation where such a genetic classification is not available.

The basic assumption is that for an animal T belonging to the genetic group g_T,

$$\beta_T = N(\mu_{g_T}, C) = \mu_{g_T} + \varepsilon_T, \quad \varepsilon_T = \text{noise}(C). \tag{13}$$

For animals with complete pedigree we assume a more specific model equation

$$\beta_T = \frac{1}{2}\beta_S + \frac{1}{2}\beta_D + \varepsilon_T^{(2)}, \quad \varepsilon_T^{(2)} = \text{noise}(C^{(2)}), \tag{14}$$

and for animals with only the sire known we assume

$$\beta_T = \frac{1}{2}\beta_S + \frac{1}{2}\mu_{g_D} + \varepsilon'_T, \quad \varepsilon'_T = \text{noise}(C'). \tag{15}$$

These equations imply relations between the group effects g_T and between the covariance matrices $C, C', C^{(2)}$. Taking the mean of (14) we find

$$\mu_{g_T} = \frac{1}{2}\mu_{g_S} + \frac{1}{2}\mu_{g_D}. \tag{16}$$

For the covariances we find

$$\begin{aligned} C &= \langle(\beta_T - \mu_{g_T})(\beta_T - \mu_{g_T})^T\rangle \\ &= \langle(\frac{1}{2}\beta_S + \frac{1}{2}\beta_D + \varepsilon_T^{(2)} - \mu_{g_T})(\frac{1}{2}\beta_S + \frac{1}{2}\beta_D + \varepsilon_T^{(2)} - \mu_{g_T})^T\rangle. \end{aligned}$$

Because of (13) and (16), this becomes

$$\begin{aligned} C &= \langle\frac{1}{2}\varepsilon_S + \frac{1}{2}\varepsilon_D + \varepsilon_T^{(2)})(\frac{1}{2}\varepsilon_S + \frac{1}{2}\varepsilon_D + \varepsilon_T^{(2)})^T\rangle \\ &= \frac{1}{4}\langle\varepsilon_S\varepsilon_S^T\rangle + \frac{1}{4}\langle\varepsilon_D\varepsilon_D^T\rangle + \langle\varepsilon_T^{(2)}\varepsilon_T^{(2)T}\rangle = \frac{1}{4}C + \frac{1}{4}C + C^{(2)}, \end{aligned}$$

assuming independence between ε_S, ε_D, and $\varepsilon^{(2)}$. Thus

$$C^{(2)} = \frac{1}{2}C.$$

Scaling equation (14) such that the error term gets covariance matrix C, we find the model equation

$$\frac{1}{\sqrt{2}}(2\beta_T - \beta_S - \beta_D) = \text{noise}(C).$$

Applying the same argument to equation (15) gives a scaled model equation

$$\frac{1}{\sqrt{3}}(2\beta_T - \beta_S - \mu_D) = \text{noise}(C).$$

By symmetry, when only the dam is known,

$$\frac{1}{\sqrt{3}}(2\beta_T - \mu_{g_D} - \beta_D) = \text{noise}(C).$$

Finally, for an animal without pedigree information derived directly from equation (13) and (16) as

$$\frac{1}{\sqrt{4}}(2\beta_T - \mu_{g_S} - \mu_{g_T}) = \text{noise}(C).$$

We can summarize all cases by writing

$$\frac{1}{\sqrt{k}}(2\beta_T - \beta_S - \beta_D) = \varepsilon_T = \text{noise}(C), \tag{17}$$

where, for unknown parents, β_S or β_D are zero, and

$$k = 4 - \text{number of known parents.}$$

Since only contrasts between genetic groups can be estimated, one can set one of the genetic group effects to zero. In particular, in the absence of genetic classification, there is only one genetic group, with zero genetic group effects, and (15) simplifies to

$$\beta_T = \frac{1}{2}\beta_S + \varepsilon'_T, \quad \varepsilon'_T = \text{noise}(C'). \tag{18}$$

This leads to following contributions to the normal equations for

$$A \begin{pmatrix} T \\ S \\ D \end{pmatrix} = \text{noise}(C) :$$

sire and dam known:

$$A^T = \left(\frac{2}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)$$

$$A^T A = \begin{pmatrix} +\frac{4}{2} & -\frac{2}{2} & -\frac{2}{2} \\ -\frac{2}{2} & +\frac{1}{2} & +\frac{1}{2} \\ -\frac{2}{2} & +\frac{1}{2} & +\frac{1}{2} \end{pmatrix}$$

sire known, dam unknown:

$$A^T = \left(\frac{2}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, 0\right)$$

$$A^T A = \begin{pmatrix} +\frac{4}{3} & -\frac{2}{3} & 0 \\ -\frac{2}{3} & +\frac{1}{3} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

sire and dam unknown:

$$A^T = \left(\frac{2}{\sqrt{4}}, 0, 0\right)$$

$$A^T A = \begin{pmatrix} \frac{4}{4} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

sire known, group of dam known:

$$A^T = \left(\frac{2}{\sqrt{3}}, -\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right)$$

$$A^T A = \begin{pmatrix} +\frac{4}{3} & -\frac{2}{3} & -\frac{2}{3} \\ -\frac{2}{3} & +\frac{1}{3} & +\frac{1}{3} \\ -\frac{2}{3} & +\frac{1}{3} & +\frac{1}{3} \end{pmatrix}$$

only group of sire and dam known:

$$A^T = \left(\frac{2}{\sqrt{4}}, -\frac{1}{\sqrt{4}}, -\frac{1}{\sqrt{4}}\right)$$

$$A^T A = \begin{pmatrix} +\frac{4}{4} & -\frac{2}{4} & -\frac{2}{4} \\ -\frac{2}{4} & +\frac{1}{4} & +\frac{1}{4} \\ -\frac{2}{4} & +\frac{1}{4} & +\frac{1}{4} \end{pmatrix}$$

group of sire known, dam unknown:

$$A^T = \left(\frac{2}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)$$

$$A^T A = \begin{pmatrix} +\frac{4}{4} & -\frac{2}{4} & 0 \\ -\frac{2}{4} & +\frac{1}{4} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

It is easy to see that the above model equations give precisely the same results as the recipe of Westell et al. (1987) without having to deal with the relationship matrix separately. Instead, only the model equations have to be set up using the above coefficients. The coefficient matrix of the normal equations is setup automatically by sweeping all model equations.

Computational considerations

Computational aspects include code complexity and execution speed. Our algorithm does not require a separate treatment of the numerator relationship matrix. Indeed, the historic problem of obtaining its inverse is completely avoided with this approach. As a consequence, only one completely general pass through the model equations is required to handle data, random effects and pedigree information.

Coding aspects

In PEST (Groeneveld et al., 1990) (version PEST2 to be released in 2018) the model equations are set up in memory with two parts 'addresses' and 'coefficients'. This is done as described in Neumaier and Groeneveld (1998), where an explicit toy example is discussed in full detail, leading to Table I.

With current cheap gigabyte memory, storing the model equations in RAM will not be problem for even the largest datasets. Once the model equations have been set up, a sweep leads directly to the coefficient matrix of the mixed model. As demonstrated above, if the pedigree model equations have been set up, this coefficient matrix will automatically include the inverse of the numerator relation matrix.

Table I: Derived matrices

dep var*		gamma	mtype	addresses				coefficients			
-0.98	-1.24	1	1	0	2	6	14	-0.16	1.00	1.00	1.00
-0.66	-	1	4	0	2	8	16	-2.96	1.00	1.00	1.00
0.38	0.59	1	1	0	4	6	18	3.24	1.00	1.00	1.00
1.54	1.00	1	1	0	4	10	20	-1.66	1.00	1.00	1.00
-0.27	-0.35	1	1	0	4	12	22	1.54	1.00	1.00	1.00
		2	2	14	24	26		1.41	-0.71	-0.71	
		2	2	16	14	26		1.41	-0.71	-0.71	
		2	2	18	14	20		1.41	-0.71	-0.71	
		2	2	20	16	14		1.41	-0.71	-0.71	
		2	2	22	18	20		1.41	-0.71	-0.71	
		2	2	24	0	0		1.00	0	0	
		2	2	26	28	0		1.15	-0.58	0	
		2	2	28	0	0		1.00	0	0	
		3	3	6				1.00			
		3	3	8				1.00			
		3	3	10				1.00			
		3	3	12				1.00			

dep var*: (dependent variable − mean)/(standard deviation)

The address and coefficient matrices in Table I correspond to MEset()%adr and MEset()%coe in the algorithm below, which form a sparse representation of the matrix A of (3) and can thus be used directly to set up the normal equations. The following gives the algorithm for this sweep, creating the coefficient matrix in sparse format as may be required for a direct solution by Cholesky decomposition or for computing the sparse inverse to get sensitivity information:

Algorithm: sweeping the model equations

Purpose: sets up the coefficient matrix of the mixed model equations
Input: complete set of model equations (as an example see table I)
Output: coefficient matrix of the mixed model equations (subroutine ebmme creates this in sparse format)

```
ME: do gamma = 1, tot_MEset
   REC: do irec = 1, MEset(gamma)%nrec
      mtype = MEset(gamma)%usecov(irec)
      do ir = 1, MEset(gamma)%nelements
         do ic = ir, MEset(gamma)%nelements
            vinv = COVset(mtype)%covinv(ir, ic)
            addon = MEset(gamma)%coe(irec, ir)*vinv*MEset(gamma)%coe(irec, ic)
            mmr = MEset(gamma)%adr(irec, ir)
            mmc = MEset(gamma)%adr(irec, ic)
            call ebmme(mmr, mmc, addon)
         end do
      end do
   end do REC
end do ME
```

If, instead, the system of equations is to be solved by conjugate gradients, the vector product $q =$

Bp as given above uses the same set of model equations and is equally compact. Again, sweeping the sets of model equations as described above covers all mixed model equations including the contributions through the inverse of the relationship matrix in (2).

Program execution - timings and scaling

A dairy cattle dataset has been used to obtain an impression on the performance of the model equation based implementation in PEST2 for the dated Gauss Seidel iteration on data and the preconditioned conjugate gradient implementation. The job was run on a €500 Linux system with a Intel Core i5-4670 3.40GHz CPU from 2013 with 16GB RAM.

The dataset consisted of more than 10 mio data records with 9 traits and many of them missing. The model had 6 covariables, 11 fixed class effects, one random and one animal factor. This resulted in more than 20 mio mixed model equations. One conjugate gradients (CG) iteration took 3.8 seconds, while the classic Jacobi/Gauss-Seidel was much slower with more than 60 seconds per iteration. Also, the number of iterations was very much smaller for CG.

Figure 1: Elapsed execution time solving one iteration by conjugate gradients on 1 through 4 cores

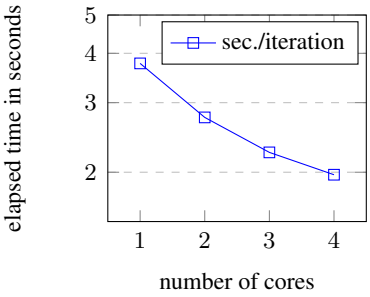

Figure 1 shows the elapsed execution time going from one to four cores. As can be seen, the speedup is substantial, although scaling is not perfect. Realistically, this is not to be expected considering the consumer level hardware with limited cache and the fact that no memory and thread placement to account for the non uniform memory access (NUMA) characteristic of the shared memory hardware effect has thus far been done. Memory requirements were around 8GB increasing by around one GB per core which is required for the private vectors in the coarse grained parallelization implemented with OpenMP (Dagum and Menon, 1998).

Outlook: inclusion of SNP data

For computational practice, it is important to note that in our normal equations, neither the relationship matrix nor its inverse occurs – hence it need not be computed at all for calculating BLUPs.

This has important implications for breeding value estimation when, in addition to pedigree information, also genomic information is present. In this case, the estimation currently proceeds with the single step gBLUP (Misztal et al., 2009), which combines genotyped with pedigree only animals in one operationally simple step. However, the method requires the inverse of a genomic relationship matrix (GRM). The GRM can be set up in a number of ways, (e.g., (VanRaden, 2007)), each including at some stage heuristics to ensure its positive definiteness. Additionally, this possibly large matrix needs to be inverted.

The animal model only became computationally feasible, after Henderson found a direct way to setup the inverse of the NRM. Such a direct procedure does not exist for the inverse of the GRM. However, as we showed in this contribution, Hendersons's mixed models can be set up solely on the basis of model equations where the relationship matrix enters neither derivation nor computations.

The fact that neither the NRM nor its inverse is needed in our formulation becomes particularly interesting where the inversion is infeasible or extremely costly as is the case in genomic relationship matrix (GRM) based gBLUP, where so far an explicit inverse is required. We are presently working on the use of the new method for the handling of genomic (SNP) data in genetic evaluation. That neither NRM and GRM is required should lead to a joint evaluation with pedigree and genomic data with no additional assumptions except the model and covariances.

References

Corbeil, R. R. and Searle, S. R. (1976). Restricted maximum likelihood (REML) estimation of variance components in the mixed model. *Technometrics*, 18:31–38.

Dagum, L. and Menon, R. (1998). OpenMP: An industry-standard api for shared-memory programming. *IEEE Comput. Sci. Eng.*, 5(1):46–55.

Fellner, W. H. (1986). Robust estimation of variance components. *Technometrics*, 28:51–60.

Fraley, C. and Burns, P. J. (1995). Large-scale estimation of variance and covariance components. *SIAM J. Sci. Comput.*, 16:192–209.

Golub, G. H. and van Loan, C. F. (2012). *Matrix Computations*, volume 3. JHU Press.

Groeneveld, E., Kovač, M., and Wang, T. (1990). PEST, a general purpose BLUP package for multivariate prediction and estimation. In *4th World Congress on genetics applied to livestock production, Edinburgh*, number XIII, pages 488–491.

Hemmerle, W. J. and Hartley, H. O. (1973). Computing maximum likelihood estimates for the mixed A.O.V. model using the W transformation. *Technometrics*, 15:819–831.

Henderson, C. (1950). Estimation of genetic parameters. *Ann. Math. Stat.*, 21:706.

Misztal, I., Legarra, A., and Aguilar, I. (2009). Computing procedures for genetic evaluation including phenotypic, full pedigree, and genomic information. *Journal of Dairy Science*, 92(9):4648–4655.

Neumaier, A. and Groeneveld, E. (1998). Restricted Maximum Likelihood Estimation of Covariances in Sparse Linear Models . *Genet. Sel. Evol.*, 1(30):3–26.

VanRaden, P. M. (2007). Efficient Methods to Compute Genomic Predictions. *J. Dairy Sci.*, 91(11):4414 – 4423.

Westell, R. A., Quaas, R. L., and Vleck, L. D. V. (1987). Genetic groups in an animal model. *Journal of Animal Science*, 71:1310–1318.

Wolfinger, R., Tobias, R., and Sall, J. (1994). Computing Gaussian likelihood and their derivatives for general linear mixed models. *SIAM J. Sci. Comput.*, 15:1294–1310.