Augmented Monte Carlo AI-REML for (co)variance component estimation in single-step models

H. Gao[1] and I. Strandén[1]

[1] Natural Resources Institute Finland (Luke), 31600 Jokioinen, Finland

Corresponding author: hongding.gao@luke.fi

Abstract

Accurate (co)variance component estimation (VCE) is essential for reliable genetic evaluation. Analytical restricted maximum likelihood (REML) estimation in single-step genomic models is often computationally demanding because mixed model equations (MME) have dense genomic relationship blocks. In this study, we extended the augmented AI-REML to an augmented Monte Carlo (MC) AI-REML approach, in which MC sampling was used to approximate the trace terms required for REML, thereby avoiding explicit inversion of the coefficient matrix of MME. The method was implemented for three single-step models: ssGBLUP, ssGTABLUP, and ssSNPBLUP. Data were simulated for 44,280 individuals over 20 generations, with 9,000 markers and three scenarios comprising 10,000, 20,000, or 30,000 genotyped individuals. Variance component estimates were consistent across single-step models and genotyping scenarios. The augmented MC AI-REML converged within 11-18 REML iterations. Computing time increased with the number of genotyped individuals for all models. Based on the current datasets, ssGBLUP was the fastest in all scenarios but showed a substantial increase in memory requirements as the number of genotyped individuals increased. ssSNPBLUP required the longest computing time. ssGTABLUP showed intermediate computing times with substantially lower memory requirements than ssGBLUP. Overall, augmented MC AI-REML provides an efficient approach for VCE in single-step models and has potential for application to large-scale genomic datasets.

Key words: Genetic parameters, ssGBLUP, ssGTABLUP, ssSNPBLUP, genomic prediction

Introduction

Genomic prediction and genetic evaluation require accurate (co)variance component estimation (VCE). Restricted maximum likelihood (REML) has long been the preferred method for performing VCE (Misztal et al., 2020). Compared with Bayesian approaches based on Gibbs sampling, REML avoids the need for Markov chain Monte Carlo (MCMC) sampling and its associated practical complications, making it an attractive and computationally efficient approach for routine VCE in genetic evaluation.

Analytical REML methods generally require factorization or inversion of the coefficient matrix of the mixed model equations (MME) to form the trace terms (Meyer, 1991; Gilmour et al., 1995; Johnson and Thompson, 1995; Jensen et al., 1997). In pedigree-based evaluations, this can be performed efficiently because the sparse inverse of the numerator relationship matrix results in sparse MME, enabling efficient computation even for large datasets. However, in single-step models, analytical REML can become computationally prohibitive because MME have dense genomic relationship blocks.

Monte Carlo (MC) REML methods, which approximate the trace terms using random samples generated from the same distribution as the observed data and the current VC estimates, can overcome the computational limitation (Matilainen et al., 2012, 2013). Recently, Bermann et al. (2025) extended MC REML to single-step genomic BLUP (ssGBLUP) by developing MC-ss-GREML. They showed that

the resulting VC estimates were in agreement with those from exact ss-GREML, while requiring only 14% of the computing time and 1% of the memory compared to the exact ss-GREML.

In this study, the aims were: (1) to extend the augmented AI-REML (Stranden et al., 2024) to augmented MC AI-REML for VCE under three single-step models (ssGBLUP, ssGTABLUP, and ssSNPBLUP); and (2) to compare the computational performance of these implementations.

Materials and Methods

Augmented MC AI-REML

In the augmented MC AI-REML, the update vector ($\boldsymbol{\Delta}$) can be obtained by solving the following augmented MME (Stranden et al., 2024):

$$\begin{bmatrix} \mathbf{C} & \mathbf{W}'\mathbf{R}^{-1}\mathbf{F} \\ \mathbf{F}'\mathbf{R}^{-1}\mathbf{W} & \mathbf{F}'\mathbf{R}^{-1}\mathbf{F} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{s}}^{*} \\ \boldsymbol{\Delta} \end{bmatrix} = \begin{bmatrix} \mathbf{W}'\mathbf{R}^{-1}\mathbf{y} \\ \mathbf{F}'\mathbf{R}^{-1}\mathbf{y} - \mathbf{t} \end{bmatrix} \quad (1)$$

where $\mathbf{C}$ is the original coefficient matrix, $\mathbf{W} = [\mathbf{X} \quad \mathbf{Z}]$ is the design matrix, $\mathbf{F}$ is the working matrix, $\mathbf{R} = \mathbf{R}_0 \otimes \mathbf{I}$, $\mathbf{R}_0$ is the residual variance covariance matrix, $\mathbf{I}$ is an identity matrix, $\otimes$ denotes the Kronecker product, $\mathbf{y}$ is the vector of observations, $\widehat{\mathbf{s}}^{*}$ is the new solution vector of fixed and random effects, and $\mathbf{t}$ is the trace vector, containing the trace contribution for each variance and covariance component in the REML score vector (i.e., the first derivatives of the REML log-likelihood).

Monte Carlo sampling was used to approximate the trace vector $\mathbf{t}$ in (1), thereby avoiding explicit inversion of the coefficient matrix. The approach follows the strategy of García-Cortés et al. (1995) and its applications (Matilainen et al., 2012, 2013). At each REML iteration, data were simulated using the current estimates of the (co)variance matrices to form the simulated phenotypes. Then simulated phenotypes were subsequently solved using the same MME as the observed data.

For the genetic part in the trace vector:

$$\mathbf{t}_{\mathrm{G}} = \mathbf{G}_0^{-1}(q\mathbf{G}_0 - \mathbf{PEV}_{\mathrm{G}})\mathbf{G}_0^{-1} \quad (2)$$

where $\mathbf{G}_0$ is the current genetic (co)variance matrix, q is the number of levels in the genetic effects, and the vector of prediction error variance (PEV) can be computed as

$$\mathbf{PEV}_{\mathrm{G}} = q\mathbf{G}_0 - \frac{1}{s}\sum_{h=1}^{s} \widehat{\tilde{\mathbf{u}}}^{h\prime}\mathbf{K}^{-1}\widehat{\tilde{\mathbf{u}}}^{h} \quad (3)$$

or

$$\mathbf{PEV}_{\mathrm{G}} = \frac{1}{s}\sum_{h=1}^{s}\left(\tilde{\mathbf{u}}^{h} - \widehat{\tilde{\mathbf{u}}}^{h}\right)'\mathbf{K}^{-1}\left(\tilde{\mathbf{u}}^{h} - \widehat{\tilde{\mathbf{u}}}^{h}\right) \quad (4)$$

where s is the number of MC samples, each $h = 1, \ldots, s$ represents one independently simulated dataset under the current VC estimates. $\tilde{\mathbf{u}}^{h}$ is the vector of simulated genetic effects for MC replicate h, and $\widehat{\tilde{\mathbf{u}}}^{h}$ is the vector of estimated genetic effects based on the simulated phenotypes. $\mathbf{K}$ is the relationship matrix used in the model, and thus $\mathbf{K} = \mathbf{H}$ for ssGBLUP model (Legarra et al., 2009; Aguilar et al., 2010; Christensen and Lund, 2010). Similar strategy can be applied for other random effects in the model to form the corresponding parts in the trace vector.

For the residual part in the trace vector:

$$\mathbf{t}_{\mathrm{R}} = \mathbf{R}_0^{-1}(n\mathbf{R}_0 - \mathbf{PEV}_{\mathrm{R}})\mathbf{R}_0^{-1} \quad (5)$$

where $\mathbf{R}_0$ is the current residual (co)variance matrix, n is the number of observations, and the vector of PEV can be computed as

$$\mathbf{PEV}_{\mathrm{R}} = n\mathbf{R}_0 - \frac{1}{s}\sum_{h=1}^{s} \hat{\tilde{\mathbf{e}}}^{h\prime}\hat{\tilde{\mathbf{e}}}^{h} \quad (6)$$

or

$$\mathbf{PEV}_{\mathrm{R}} = \frac{1}{s}\sum_{h=1}^{s}\left(\tilde{\mathbf{e}}^{h} - \hat{\tilde{\mathbf{e}}}^{h}\right)'\left(\tilde{\mathbf{e}}^{h} - \hat{\tilde{\mathbf{e}}}^{h}\right) \quad (7)$$

where $\tilde{\mathbf{e}}^{h}$ is the vector of simulated residuals for MC replicate h, $\hat{\tilde{\mathbf{e}}}^{h}$ is the vector of estimated residuals. A detailed description of the computational strategy can be found in Bermann et al., (2025).

In this study, a MC sample size of five was used. Thus, at each REML iteration, the original MME were solved six times (once for the observed data and five times for the simulated data) and the augmented MME were solved once. Following (Matilainen et al., 2019), the same MC samples were reused across REML iterations to reduce the MC noise. Convergence of the augmented MC AI-REML was assessed based on the round-to-round change in VC estimates, with the convergence threshold value set to 1.0E-10. All analyses were conducted

using the MiX99 software (Pitkänen et al., 2022).

Single-step models

Three single-step models (ssGBLUP, ssGTABLUP, and ssSNPBLUP) were applied for VCE. In all models, genomic and pedigree information were blended to account for residual polygenic effects with $w = 0.1$.

1) ssGBLUP (Legarra et al., 2009, Christensen and Lund, 2010)

In ssGBLUP, the genomic relationship matrix (GRM) was built using Method 1 from VanRaden (2008).

$$\mathbf{G} = (1-w)\mathbf{Z}\mathbf{B}\mathbf{Z}' + w\mathbf{A}_{22} \quad (8)$$

where $\mathbf{Z}$ is the centered marker matrix of size $n \times m$, defined as $\mathbf{Z} = \mathbf{M} - 2\mathbf{P}$, and $\mathbf{M}$ is original $n \times m$ marker covariates matrix (coded as 0,1,2), $\mathbf{P}$ is an $n \times m$ centering matrix, defined as $\mathbf{P} = \mathbf{1}\mathbf{p}'$, and $\mathbf{p}$ is a vector of allele frequencies of base population, m is the number of markers, n is the number of genotyped individuals. $\mathbf{B}$ is a scaling matrix of size $m \times m$, defined as $\mathbf{B}^{-1} = k\mathbf{I}$, with $k = \sum_{l=1}^{m} 2p_l(1-p_l)$, $\mathbf{A}_{22}$ is the pedigree relationship matrix among the genotyped individuals. Allele frequencies in the base population were estimated from all available genomic data using the Bpop software (Strandén and Mäntysaari, 2020). The inverse of the GRM ($\mathbf{G}^{-1}$) was precomputed and stored in memory.

2) ssGTABLUP (Vandenplas et al., 2023)

The ssGTABLUP model used Woodbury matrix identity to avoid explicitly building $\mathbf{G}$ and $\mathbf{G}^{-1}$. The $\mathbf{G}^{-1}$ can be decomposed as

$$\mathbf{G}^{-1} = \frac{1}{w}\mathbf{A}_{22}^{-1} - \frac{1}{w}\mathbf{A}_{22}^{-1}\mathbf{Z}\mathbf{K}^{-1}\mathbf{Z}'\frac{1}{w}\mathbf{A}_{22}^{-1} \quad (9)$$

where $\mathbf{K} = \frac{1}{w}\mathbf{Z}'\mathbf{A}_{22}^{-1}\mathbf{Z} + \frac{1}{1-w}\mathbf{B}^{-1}$ and was precomputed and stored in memory.

3) ssSNPBLUP (Liu et al., 2014)

The ssSNPBLUP model directly operated on the marker covariates matrix ($\mathbf{M}$) and thus required no precomputation.

Data

Data were simulated using AlphasimR (Gaynor et al., 2021) for a single trait with genetic and residual variances of $\sigma_u^2 = 50$ and $\sigma_e^2 = 50$, respectively, in the base population. The breeding scheme was as follows: in each generation, 2,000 females and 200 males were selected based on their phenotypes and randomly mated with each selected female producing one offspring. The simulated population comprised 44,280 individuals with single record each across 20 generations. For VCE, three genotyping scenarios were investigated, in which 10,000, 20,000, or 30,000 youngest individuals were genotyped for 9.000 markers.

Results & Discussion

Variance component estimates were consistent across models and genotyping scenarios (Table 1). As expected, the estimates differed from the initial simulated parameters because phenotypic selection over 20 generations reduced the genetic variance. Moreover, in single-step models, additive genetic variance is estimated on the scale of the combined relationship matrix (**H**) rather than on the scale of the pedigree-based relationship matrix (**A**). The VC estimates were also in agreement with those reported by Gao et al. (2026), in which analytical and MC EM-REML were applied to the same datasets.

Across all analyses, augmented MC AI-REML converged within 11-18 REML iterations. The number of iterations required showed only minor variation among models and genotyping scenarios. In contrast, MC EM-REML required approximately 200-300 REML iterations based on the same datasets (Gao et al., 2026). This difference is consistent with the faster convergence expected for AI-REML, which uses both first (gradient) and second order derivative (curvature) information of the log likelihood compared with the slower but generally more robust EM-REML algorithm.

The computational efficiency of MC AI-REML for ssGBLUP model was also shown by Bermann et al. (2025), who reported that MC-ss-GREML required only 14% of the computing time and 1% of the memory compared to the exact ss-GREML while producing similar VC estimates. The substantial reductions in computing time and memory were attributed mainly to avoiding storage and factorization of the coefficient matrix of MME. Their findings demonstrated the potential of MC AI-REML for computationally efficient VCE in large single-step genomic models.

Computing time increased with the number of genotyped individuals for all models. Substantial differences in computational performance were observed among the models (Table 1). ssGBLUP required the least computing time in all scenarios, whereas ssSNPBLUP was the most computationally demanding model. ssGTABLUP showed intermediate computing times between ssGBLUP and ssSNPBLUP. This ranking was expected given the relatively small numbers of genotyped individuals in the present study. However, as the number of genotyped individuals increased, computing time increased proportionally more for ssGBLUP than for ssGTABLUP or ssSNPBLUP, indicating less favorable computational scaling of ssGBLUP. This pattern is consistent with the computational complexity of breeding value estimation, which scale as $O(n^2)$ for ssGBLUP but approximately as $O(mn)$ for ssGTABLUP and ssSNPBLUP (Vandenplas et al., 2023).

Peak memory usage for ssGBLUP increased sharply with the number of genotyped individuals, whereas the increase was much more moderate for ssGTABLUP and ssSNPBLUP. The high memory requirement of ssGBLUP was mainly due to the precomputation and in-memory storage of the dense $\mathbf{G}^{-1}$. In contrast, ssGTABLUP and ssSNPBLUP substantially reduced memory requirements, although at the expense of increased computing time. Because these methods work on marker covariates matrix. Based on the current datasets, ssGTABLUP provided a compromise between computing time and memory usage, particularly as the number of genotyped individuals increased.

Table 1: Estimated variance components (standard errors), number of REML iterations (N), REML computing time, and peak RAM usage across single-step models and genotyping scenarios

Data[1]	M[2]	σ_u^2	σ_e^2	N	Time	RAM
10	G	35.0 (0.97)	52.2 (0.59)	14	3m18s	2.4
	T	34.1 (0.95)	52.6 (0.59)	14	12m7s	2.2
	S	34.1 (0.95)	52.6 (0.59)	18	39m54s	1.6
20	G	34.7 (0.93)	52.9 (0.52)	11	7m:4s	7.1
	T	34.2 (0.92)	53.3 (0.52)	11	20m9s	2.9
	S	34.2 (0.92)	53.3 (0.52)	18	1h54m	2.3
30	G	36.8 (0.98)	53.0 (0.47)	13	19m30s	15.5
	T	36.6 (0.97)	53.2 (0.47)	12	28m59s	3.6
	S	36.6 (0.97)	53.2 (0.47)	12	1h45m	2.9

[1] 10, 20, and 30 indicate 10,000, 20,000, and 30,000 genotyped individuals, respectively.
[2] G, T, and S indicate ssGBLUP, ssGTABLUP, and ssSNPBLUP, respectively.

Peak memory usages were similar between augmented MC AI-REML and MC EM-REML for the corresponding single-step models (Gao et al., 2026). This was expected because memory usage was determined primarily by the implementation of each single-step model rather than by the REML optimization algorithm. In both studies, memory requirements increased with the number of genotyped individuals for ssGBLUP, whereas they remained considerably lower for ssGTABLUP and ssSNPBLUP. These results suggest that augmented MC AI-REML retains the memory advantages of MC-based VCE while converging substantially faster than MC EM-REML.

Conclusions

We extended augmented AI-REML to augmented MC AI-REML for VCE under three single-step models: ssGBLUP, ssGTABLUP, and ssSNPBLUP. All models produced similar VC estimates within relatively few REML iterations. Computational performance differed among the single-step models. ssGTABLUP offered a better trade-off between computing time and memory requirements. Overall, augmented MC AI-REML provides an efficient approach for VCE in single-step models and has potential for application to large-scale genomic datasets.

References

Aguilar, I., I. Misztal, D. L. Johnson, A. Legarra, S. Tsuruta, and T. J. Lawlor. 2010. Hot topic: a unified approach to utilize phenotypic, full pedigree, and genomic information for genetic evaluation of Holstein final score. *J Dairy Sci.* 93(2):743-752.

Bermann, M., A. Legarra, I. Aguilar, A. Alvarez-Munera, I. Misztal, and D. Lourenco. 2025. Estimation of (co)variance components for very large datasets and complex single-step genomic models. *Genet. Sel. Evol.* 57(1):63.

Christensen, O. F. and M. S. Lund. 2010. Genomic prediction when some animals are not genotyped. *Genet Sel Evol.* 42:2.

Gao, H., M. Taskinen, T. J. Pitkänen, A. Ylitalo, N. Vargas Jurado, M. Lidauer, and I. Strandén. 2026. Estimation of variance components in single-step models using MiX99. *in Proc. 13th World Congr. Genet. Appl. Livest. Prod.*, Wisconsin, USA.

García-Cortés, L., C. Moreno, L. Varona, and J. Altarriba. 1995. Estimation of prediction-error variances by resampling. *J Anim Breed Genet.* 112(1-6):176-182.

Gaynor, R. C., G. Gorjanc, and J. M. Hickey. 2021. AlphaSimR: an R package for breeding program simulations. *G3* 11(2).

Gilmour, A. R., R. Thompson, and B. R. Cullis. 1995. Average information REML: An efficient algorithm for variance parameter estimation in linear mixed models. *Biometrics* 51(4):1440-1450.

Jensen, J., E. A. Mäntysaari, P. Madsen, and R. Thompson. 1997. Residual maximum likelihood estimation of (co) variance components in multivariate mixed linear models using average information. *J Indian Soc Agric Stat.* 49:215-236.

Johnson, D. L. and R. Thompson. 1995. Restricted Maximum-Likelihood-Estimation of Variance-Components for Univariate Animal-Models Using Sparse-Matrix Techniques and Average Information. *J Dairy Sci.* 78(2):449-456.

Legarra, A., I. Aguilar, and I. Misztal. 2009. A relationship matrix including full pedigree and genomic information. *J Dairy Sci.* 92(9):4656-4663.

Liu, Z., M. E. Goddard, F. Reinhardt, and R. Reents. 2014. A single-step genomic model with direct estimation of marker effects. *J Dairy Sci.* 97(9):5833-5850.

Matilainen, K., E. A. Mantysaari, M. H. Lidauer, I. Stranden, and R. Thompson. 2012. Employing a Monte Carlo algorithm in expectation maximization restricted maximum likelihood estimation of the linear mixed model. *J Anim Breed Genet.* 129(6):457-468.

Matilainen, K., E. A. Mantysaari, M. H. Lidauer, I. Stranden, and R. Thompson. 2013. Employing a Monte Carlo algorithm in Newton-type methods for restricted maximum likelihood estimation of genetic parameters. *PLoS One* 8(12):e80821.

Matilainen, K., E. A. Mantysaari, and I. Stranden. 2019. Efficient Monte Carlo algorithm for restricted maximum likelihood estimation of genetic parameters. *J Anim Breed Genet.* 136(4):252-261.

Meyer, K. 1991. Estimating Variances and Covariances for Multivariate Animal-Models by Restricted Maximum-Likelihood. *Genet. Sel. Evol.* 23(1):67-83.

Misztal, I., D. Lourenco, and A. Legarra. 2020. Current status of genomic evaluation. *J Anim Sci.* 98(4).

Pitkänen, P. J., H. Gao, A. Kudinov, M. Taskinen, E. A. Mäntysaari, M. H. Lidauer, and I. Strandén. 2022. From data to genomic breeding values with the MiX99 software suite. *in Proc. 12th World Congr. Genet. Appl. Livest. Prod.,* Rotterdam, The Netherlands.

Strandén, I. and E. A. Mäntysaari. 2020. Bpop: an efficient program for estimating base population allele frequencies in single and multiple group structured populations. *Agric. Food Sci.* 29(3):166–176.

Stranden, I., E. A. Mantysaari, M. H. Lidauer, R. Thompson, and H. Gao. 2024. A computationally efficient algorithm to leverage average information REML for (co)variance component estimation in the genomic era. *Genet. Sel. Evol.* 56(1):73.

Vandenplas, J., J. Ten Napel, S. N. Darbaghshahi, R. Evans, M. P. L. Calus, R. Veerkamp, A. Cromie, E. A. Mantysaari, and I. Stranden. 2023. Efficient large-scale single-step evaluations and indirect genomic prediction of genotyped selection candidates. *Genet. Sel. Evol.* 55(1):37.

VanRaden, P. M. 2008. Efficient methods to compute genomic predictions. *J Dairy Sci.* 91(11):4414-4423.