# PX × AI : ALGORITHMICS FOR BETTER CONVERGENCE IN RESTRICTED MAXIMUM LIKELIHOOD ESTIMATION

**Karin Meyer**

Animal Genetics and Breeding Unit, University of New England (a joint venture with NSW Agriculture), Armidale, NSW 2351, Australia

## INTRODUCTION

Maximising the (log) likelihood ($\log \mathcal{L}$) in restricted maximum likelihood (REML) estimation of variance components almost invariably represents a constrained optimisation problem. Iterative algorithms available to solve this problem differ substantially in computational resources needed, ease of implementation, sensitivity to choice of starting values and rates of convergence. One of the most widely used methods is the 'average information' (AI) algorithm, which "often converges in a few rounds of iteration" (Thompson *et al.* 2005). However, there have been some, albeit mainly anecdotal reports of the AI algorithm failing to converge, in particular for analyses involving multiple random effects, numerous traits or 'bad' starting values. A popular alternative are expectation-maximisation (EM) algorithms. While these are guaranteed to increase $\log \mathcal{L}$ in each iterate, they are often painfully slow to converge. Recently, Foulley and van Dyk (2000) considered the 'parameter expanded' (PX) variant of the EM algorithm for mixed model REML, and demonstrated dramatically improved convergence compared to standard EM. Yet, there has been little use of the PX-EM algorithm. No comparisons between AI and PX-EM algorithms are available. This paper compares convergence rates of standard EM, PX-EM and AI algorithms for some practical examples of analyses of beef cattle data.

## ALGORITHMS

**Average Information.** In essence, the AI algorithm is a modified Newton-Raphson procedure, replacing second derivatives of $\log \mathcal{L}$ with the average of their observed and expected values. Like other second order algorithms, it is expected to have quadratic convergence, but it shares their drawbacks. Firstly, estimates are not constrained to the parameter space. This can be overcome by an appropriate parameterisation, e.g. by estimating the elements of the Cholesky factor of a covariance matrix rather than the covariances, often teamed with a log transformation of the diagonal elements (Pinheiro and Bates 1996). Secondly, $\log \mathcal{L}$ is not guaranteed to increase. While the AI matrix is generally positive definite, which should yield an increase in $\log \mathcal{L}$, it is prone to 'overshooting', which can yield a decrease. Often step size modification are necessary to ensure $\log \mathcal{L}$ attains a maximum. Typically, these require additional computations.

**Expectation-Maximisation.** On the other hand, EM type algorithms generally have monotone convergence, i.e. $\log \mathcal{L}$ increases in each iterate, and yield estimates of covariance components within the parameter space. Moreover, they are easier to implement and require less memory than corresponding AI algorithms. However, EM algorithms utilise information from first derivatives of the likelihood only, and thus are expected to converge linearly. This can be very slow, and may require many rounds of iteration. This behaviour has motivated numerous attempts to speed up convergence. Modifications suggested include simple predictions of future values from estimates in previous iterates, such as 'accelerated EM' (Laird *et al.* 1987), Quasi-Newton type schemes, and generalised EM algorithms; see Meng and Van Dyk (1997) for a review.

**Parameter expansion.** Probably the most interesting among the new, 'fast' EM procedures is the PX-EM algorithm proposed by Liu *et al.* (1998). Consider the standard linear, mixed model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \mathbf{e} \qquad \text{with} \quad \text{Var}(\mathbf{u}) = \boldsymbol{\Sigma} \otimes \mathbf{A} \tag{1}$$

where $\mathbf{y}$, $\boldsymbol{\beta}$, $\mathbf{u}$ and $\mathbf{e}$ are the vectors of observations, fixed effects, random effects and residuals, respectively, $\mathbf{X}$ and $\mathbf{Z}$ denote the corresponding incidence matrices, and $\boldsymbol{\Sigma}$ is the matrix of covariances between random effects to be estimated. For PX-EM, rewrite (Eq. 1) as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}(\mathbf{I} \otimes \boldsymbol{\alpha})\mathbf{u}^{\star} + \mathbf{e} \qquad \text{with} \quad \text{Var}(\mathbf{u}^{\star}) = \boldsymbol{\Sigma}^{*} \otimes \mathbf{A} \tag{2}$$

The elements of $\boldsymbol{\Sigma}^{*}$ are then estimated assuming $\boldsymbol{\alpha} = \mathbf{I}$, i.e. as for standard EM. In addition, we estimate the elements of $\boldsymbol{\alpha}$. If there are $q$ traits (or random regression coefficients), there are up to $q^2$ additional parameters. Estimators are obtained in standard fashion, equating first derivatives of the expectation of the complete data likelihood of $\mathbf{y}$ (assuming $\boldsymbol{\beta}$ and $\mathbf{u}$ are known) with respect to the elements of $\boldsymbol{\alpha}$ to zero, and solving the resulting system of equations; see Foulley and van Dyk (2000) for details. Estimates of $\boldsymbol{\Sigma}$ are then obtained applying the reduction function $\boldsymbol{\Sigma} = \boldsymbol{\alpha}\boldsymbol{\Sigma}^{*}\boldsymbol{\alpha}'$. Finally, residual covariances are estimated as in standard EM, but adjusting for the current estimate of $\boldsymbol{\alpha} \neq \mathbf{I}$ in calculating residuals, i.e. using $\mathbf{e} = \mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}(\hat{\boldsymbol{\alpha}} \otimes \mathbf{I})\mathbf{u}^{\star}$.

For most algorithms, computational requirements of REML estimation increase with the number of parameters, both per iterate and overall. Hence, it seems counter-intuitive to expand their number. Loosely speaking, the efficacy of PX-EM can be attributed to the extra parameters capturing 'information' which is not utilised in EM. In each iterate, we treat the current parameter values as if they maximised $\log \mathcal{L}$. Hence, away from the maximum, the expectation of the complete likelihood is computed with error. The deviation of $\hat{\boldsymbol{\alpha}}$ from $\mathbf{I}$ gives a measure of the error. Adjusting estimates of $\boldsymbol{\Sigma}$ for $\hat{\boldsymbol{\alpha}}$ then can be thought of as regressing estimates on the difference between $\hat{\boldsymbol{\alpha}}$ and its assumed value of $\mathbf{I}$ in standard EM (Liu *et al.* 1998).

## MATERIAL AND METHODS

**Data.** Three examples of animal model analyses of beef cattle data were considered (Table 1). Case A (Meyer and Kirkpatrick 2005) comprised 4 live ultra-sound scanning measures, treating records on males and females as different traits. This yielded an eight-variate analysis, fitting a simple animal model and, with zero residual covariances between sexes, 56 parameters. For case B, birth, weaning and yearling weights for cattle in a large herd were analysed together, fitting genetic and permanent environmental (PE) maternal effects for all three traits. Finally, case C (Meyer *et al.* 2004) involved 2 traits, mature cow weight and gestation length. With repeated records for trait 1, a PE effect of the animal was fitted as random effect (RE). Trait 2 was considered a trait of the calf, affected by both PE and genetic effects of the dam. This gave 4 REs in the model and 9 covariances to be estimated.

**Table 1.** Characteristics of examples

| | Number of | | | | |
|---|---|---|---|---|---|
| | Traits | Records | Anim.s | Equat.s | RE | Par.s |
| A | 8 | 20 171 | 8 044 | 65 030 | 1 | 56 |
| B | 3 | 10 479 | 6 247 | 31 252 | 3 | 24 |
| C | 2 | 32 303 | 66 169 | 197 040 | 4 | 9 |

**Analyses.** REML estimates of variance components were obtained employing AI, EM, PX-EM and a combination of PX-EM and AI (PX×AI) algorithms for the same starting values. AI was reparameterised to the elements of the Cholesky factors of the covariance matrices to

**Table 2.** Convergence characteristics (brackets denote failure to converge)

| | | | Case A | | Case B | | Case C | |
|---|---|---|---|---|---|---|---|---|
| | | | 'Good' | 'Bad' | 'Good' | 'Bad' | 'Good' | 'Bad' |
| log $\mathcal{L}$ | Start | 0 | -391.82 | -3406.69 | -2218.65 | -6261.90 | -39.20 | -2656.59 |
| | EM | 20 | -18.29 | -89.02 | -46.31 | -91.34 | -1.15 | -80.39 |
| | | 50 | -14.92 | -74.82 | -30.99 | -64.42 | -0.92 | -48.21 |
| | | 100 | -11.16 | -58.49 | -16.76 | -36.14 | | |
| | | 500 | – | – | -6.50 | -3.68 | – | – |
| | PX-EM | 20 | -3.71 | -5.73 | -12.46 | -32.63 | -1.53 | -4.13 |
| | | 50 | -2.10 | -2.21 | -7.33 | -19.85 | -1.43 | -1.46 |
| | | 100 | -2.03 | -2.02 | -3.56 | -10.36 | | -0.95 |
| | | 500 | – | – | -0.43 | -1.37 | – | – |
| No. of | AI | | 12 | (10) | (2) | 42 | (4) | (36) |
| iterates | | log | 13 | (24) | (12) | (8) | | |
| | PX×AI | a | 2+6 | 2+7 | 3+7 | 3+9 | | |
| | | b | 4+6 | 4+7 | 5+8 | 5+7 | | |
| | | c | 6+5 | 6+6 | – | – | | |

be estimated, and AI-log took logarithmic values of the diagonal elements in addition. Both enforced an increase in log $\mathcal{L}$ at each step. The PX×AI algorithm involved a small number (2–6) of PX-EM iterates, followed by AI. "Good" and "bad" starting values were considered for each case. All computations were carried out using our REML program WOMBAT (Meyer 2006).

## RESULTS AND DISCUSSION

Numbers of iterates required for the AI based, and values of log $\mathcal{L}$ (as deviations from the maximum) for the EM type algorithms at selected iterates are given in Table 2. Changes in log $\mathcal{L}$ for early iterates are shown in Figures 1 and 2 for cases B and C, respectively. Overall, results confirmed the slow convergence rates of the EM type algorithms. Performance of the PX-EM algorithm in the first few iterates was generally at least as good as that of the AI algorithm, and substantially better if AI (or AI-log) struggled to improve log $\mathcal{L}$. However, for later iterates, PX-EM shared the slow convergence of standard EM, its advantage over EM predominantly due

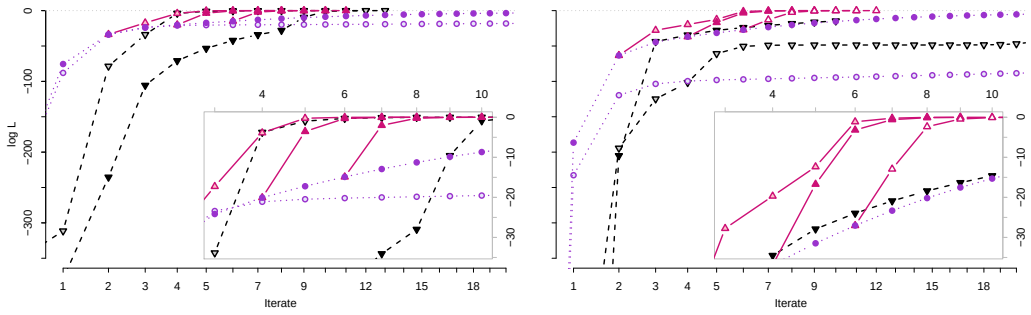


**Figure 1.** Early iterates for case A for good (left) and bad (right) starting values, using AI (▽), AI-log (▼), EM (○), PX-EM (●), and PX-AI (▲) algorithms (inset : iterates 3–10 enlarged).

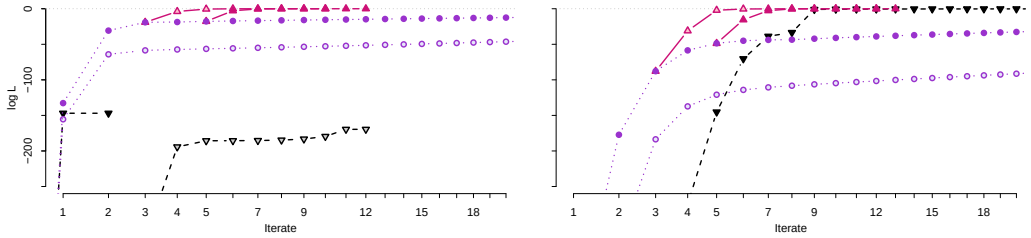


**Figure 2.** Early iterates for case B (as Figure 1 otherwise).

to its performance in the first few steps. Convergence of PX-EM in later iterates might be improved by applying an acceleration technique. The AI algorithm was found to be fairly sensitive to starting values and choices involved. In terms of parameterisation, AI-log was less affected by convergence problems at the boundary of the parameter space than AI, but tended to require more iterates. Step size modifications were usually chosen to increase $\log \mathcal{L}$ as much as possible in a given step, but enforcing just an increase often worked just as well and required less computations. No indicators as to which strategy was less likely to 'get stuck' somewhere below the maximum could be identified. Clearly, case C represented a model where the shape of the likelihood surface was not conducive to the AI algorithm.

Combining a few, initial iterates of PX-EM with AI in subsequent iterates proved highly effective. In most instances, the PX-EM algorithm yielded 'starting points' for AI sufficiently close to the maximum of $\log \mathcal{L}$ so that AI converged rapidly, even if AI (or AI-log) for the same starting values had failed. Results indicate that the PX×AI algorithm would be advantageous for routine REML estimation. A similar, but unsubstantiated suggestion has been made by Thompson *et al.* (2005). While no 'cure-all', the PX×AI algorithm seemed especially useful for reducing computational demands of analyses involving many traits or multiple random effects.

## CONCLUSIONS

More reliable and often faster convergence of REML estimation can be achieved by combining algorithms : Exploit the stability and good performance of the PX-EM algorithm in the first few iterates, then switch to the AI algorithm with rapid convergence close to the maximum of $\log \mathcal{L}$.

## REFERENCES

Foulley, J. L. and van Dyk, D. A. (2000) *Genet. Select. Evol.* **32**:143–163.
Laird, N., Lange, N. and Stram, D. (1987) *J. Amer. Stat. Ass.* **82**:97–105.
Liu, C., Rubin, D. B. and Wu, Y. N. (1998) *Biometrika* **85**:755–770.
Meng, X.-L. and Van Dyk, D. (1997) *J. Roy. Stat. Soc. B* **59**:511–567.
Meyer, K. (2006) *Proc. Eigth World Congr. Genet. Appl. Livest. Prod.* Comm. No. 27–00.
Meyer, K., Johnston, D. J. and Graser, H.-U. (2004) *Austr. J. Agric. Res.* **55**:195–210.
Meyer, K. and Kirkpatrick, M. (2005) *Genet. Select. Evol.* **37**:1–30.
Pinheiro, J. C. and Bates, D. M. (1996) *Stat. Comp.* **6**:289–296.
Thompson, R., Brotherstone, S. and White, I. M. S. (2005) *Phil. Trans. R. Soc. B* **360**:1469–1477.

## ACKNOWLEDGEMENTS

This work was supported by Meat and Livestock Australia **(www.mla.com.au)** and the International Livestock Resources and Information Centre **(www.ilric.com)**.