ORDERING STRATEGIES TO REDUCE COMPUTATIONAL REQUIREMENTS IN VARIANCE COMPONENT ESTIMATION

Karin Meyer

Animal Genetics and Breeding Unit[1], University of New England, Armidale, NSW 2351

SUMMARY

Computational requirements for sparse matrix factorisation or inversion are highly dependent on the 'fill-in' created. This can be reduced by judicious re-ordering of equations. It is shown that use of newer ordering strategies, with corresponding computer code available in the public domain, can reduce the time required for ordering and computational requirements of analyses dramatically.

Keywords: Mixed models, variance components estimation, sparse matrix factorisation, ordering

INTRODUCTION

Mixed model analyses of data from livestock improvement schemes generally involve manipulations of large, sparse matrices. In particular, estimation of variance components via restricted maximum likelihood (REML) requires the Cholesky decomposition or the inverse of the coefficient matrix in the mixed equations for each likelihood evaluation. Computational steps for this can be thought of as sequentially 'absorbing' one row and column into the remainder of the matrix. Clearly, the number of calculations required for each of these steps increases quadratically with the number of non-zero off-diagonal elements in the row. Moreover, each step is likely to create additional, non-zero entries in the remaining rows and columns, commonly referred to as 'fill-in'. In turn, the amount of 'fill-in' determines computational requirements of subsequent steps.

Judicious choice of the order in which to process rows and columns is thus critical for computational efficiency. A number of general ordering strategies, based upon graph theory, exist and various strategies are readily implemented using software available in the public domain. This paper examines the impact of different ordering strategies on computational demands of variance component analyses of beef cattle data.

MATERIAL AND METHODS

Fill-in created and number of operations required in the Cholesky decomposition of the mixed model matrix were examined for 7 examples, summarised in Table 1. Cases A to D represented uni- and bivariate analyses of traits measured on Hereford cattle (Meyer et al. 2004). Case A involved repeated records for mature cow weights (MCW), and fitted genetic and permanent environmental effects of the animal. Case B treated gestation length (GL) as a record of the calf, and allowed for both genetic and permanent environmental maternal effects. Case C comprised a bivariate analysis of birth (BW) and weaning (WW) weight records. Case D was an analysis of BW together with days to calving (DC). In addition to direct genetic effects, both maternal effects were fitted for BW and WW. DC involved repeated records per animal and was thus modelled fitting animals' genetic and permanent environmental effects. Cases E and F were random regression analyses of weights of Angus cattle (Meyer 2005a), fitting quadratic and quartic regressions on Legendre polynomials of age for animals' direct, genetic

[1]AGBU is a joint venture of NSW Department of Primary Industries and the University of New England

Table 1. Characteristics of the analyses examined

Scenario	Case	Traits	Number of				
			records	animals[A]	animals[B]	equations[C]	elements[C]
Univariate	A	MCW	9,850	6,886	18,274	25,592	113,505
	B	GL	33,006	33,006	93,922	214,518	1,267,769
Bivariate	C	BW, WW	68,408	39,036	60,537	282,949	4,919,953
	D	BW, DC	102,431	75,484	116,894	391,621	4,085,122
Multivariate	E	WT : RR-3	84,534	20,731	47,463	384,819	6,133,601
	F	WT : RR-5	84,534	20,731	47,463	521,211	10,350,163
	G	SCAN-8	262,872	74,268	103,467	837,915	28,592,845

[A] in the data, [B] in total, [C] in the mixed model matrix

and permanent environmental effects, respectively, and corresponding quadratic regressions for maternal effects. Case G was the eight-trait multivariate analysis of scan traits recorded on Angus cattle, considered by Meyer (2005b).

A symbolic factorisation of the mixed model matrix for each analysis was carried out, using permutations determined by several ordering strategies. Operations required in the corresponding Cholesky factorisation were counted, defining an operation as a pair of floating point calculations, consisting of one addition/subtraction and one multiplication/division. Orderings using the multiple minimum degree procedure were obtained using routine GENMMD (George and Liu 1981; Liu 1985). Approximate minimum degree (AMD), AMD with a pre-ordering to remove dense rows (QAMD), and approximate minimum fill (AMF) orderings were determined by routines DMUMPS_197, DMUMPS_421 and DMUMPS_337, respectively, extracted from the MUMPS package (Amestoy et al. 1998, 2001). Routine METIS_NodeND from the METIS package (Karypis and Kumar 1998b) provided multilevel nested dissection orderings. In the following, Ma, b denotes a METIS ordering, obtained using a graph separators in each dissection step, and considering vertices with degree b times higher than average as dense, placing them at the bottom of the graph. All computations were carried out on a Compaq 64-bit Alpha station with a processor rated at 667 Mhz.

RESULTS

Results are summarised in Tables 2 and 3. Not surprisingly, there was a strong relationship between numbers of non-zero elements and operation counts. On the whole, AMD and QAMD orderings were comparable to MMD, but quicker to obtain. In all cases, METIS provided substantially better orderings than the minimum degree procedures, and did so at a fraction of the time required. Similar results have been reported by Ducrocq and Druet (2003). In contrast, for smaller test cases (not shown), MMD and (Q)AMD performed much better than METIS. Considering more graph separators per step generally increased the quality of ordering, especially for the larger and more complex analyses, but increased ordering times dramatically. Disregarding very dense rows in ordering tended to increase ordering time, and was only advantageous for cases A and E. AMF provided orderings with fill-ins comparable to those from METIS, but generally higher operation counts. For case B, computing times for one likelihood evaluation were 153 and 273 secs for M1,0 and MMD orderings, respectively. Corresponding times for one average information (AI) REML iterate, as implemented in DFREML (Meyer 1998), were 946 and

Table 2. Number of non-zero elements (NNZ) and operations (NOPS) for uni/bivariate analyses

	Case A			Case B			Case C			Case D		
Method	NNZ	NOPS^A	t^B	NNZ^A	NOPS^C	t^B	NNZ^A	NOPS^C	t^B	NNZ^A	NOPS^C	t^B
MMD	692,112	108.68	4.0	20.06	27.47	26.1	37.80	46.68	589	46.95	57.78	454
AMD	686,529	105.32	3.9	19.60	24.82	8.3	38.91	51.46	404	47.83	63.82	703
QAMD	679,395	103.61	0.4	21.32	30.44	6.3	36.77	43.01	16	43.13	46.95	25
AMF	550,706	57.31	5.9	15.92	16.54	10.5	32.82	29.71	555	35.50	28.87	976
M1,0	569,066	48.97	0.9	16.40	14.15	8.5	33.10	22.90	22	40.27	30.29	72
M5,0	593,276	54.71	3.2	17.00	15.99	37.7	30.23	17.86	56	35.98	21.91	305
M5,20	576,024	50.43	4.9	17.32	16.52	41.0	32.86	21.22	83	36.42	21.99	145
M10,0	569,274	48.48	5.7	16.47	14.74	68.6	30.14	17.65	96	35.32	20.67	603
M10,20	561,245	47.29	9.6	16.41	14.59	79.9	32.36	20.39	147	36.37	22.53	263
M12,0	553,376	45.36	6.9	16.33	14.34	84.0	30.00	17.76	112	34.87	20.65	715
M14,0	577,063	51.07	7.7	16.44	14.69	98.2	30.45	17.88	82	34.69	20.09	840

Shaded entries show 'best' values; ^{A} $\times 10^6$, ^{B} time for ordering & symbolic factorisation (secs); ^{C} $\times 10^9$

1634 secs. Times for case D were 1895 (M14,0) *vs.* 4458 (MMD) secs per AI iterate and 308 *vs.* 676 secs per likelihood, with memory requirements of 619 *vs.* 812 Mbytes. Hence reductions in operation counts translated almost directly into proportionally reduced computing times.

DISCUSSION

Computer code for all ordering schemes considered requires the same format of the adjacency structure of the sparse matrix to be factored. Hence implementation of alternatives to GENMMD, used widely so far, is straightforward. No strategy proved best for all examples considered. Times required for ordering were sufficiently small, in particular for QAMD and METIS considering 5 or less graph separators, that a number of alternatives could readily be trialled. Especially for large analyses, computational savings achieved outweighed the additional effort for ordering by orders of magnitude.

The multilevel, nested dissection implemented in METIS is an example of a modern scheme combining top-down and bottom-up ordering strategies. In essence, METIS attempts to recursively partition the graph into independent subgraphs. Separators and dense vertices are ordered to the bottom of the graph. When subgraphs reach a minimum size, a minimum degree ordering is applied to the remainder. This 'divide and conquer' technique appeared to be well suited to the structure of mixed model equations considered. Herds provided a natural separator, as animals effects, in part at least, and contemporary groups were nested within herds. Other strategies exist which claim a tighter coupling between steps and better performance (e.g. Pellegrini et al. 2000; Schulze 2001). Different heuristics to the minimum degree or minimum fill have been shown to improve orderings in other cases (Ng and Raghavan 1999). Further research should examine these alternatives. Operation counts are computationally inexpensive to obtain. Hence, it may be feasible to apply a global optimisation strategy to permute orderings derived from a standard strategy like METIS to reduce operation counts further (Tier, 2003; *pers. comm.*).

CONCLUSIONS

The multilevel nested dissection scheme of Karypis and Kumar (1998a) is well suited to ordering sparse

Table 3. Number of non-zero elements (NNZ) and operations (NOPS) for multivariate analyses

Case E				Case F				Case G			
Method	NNZ^A	NOPS^B	t^C	Method	NNZ^A	NOPS^B	t^C	Method	NNZ^A	NOPS^B	t^C
MMD	84.33	188.87	498	MMD	133.40	382.30	754	MMD	231.37	630.83	8408
AMD	81.62	174.40	163	AMD	135.77	388.13	204	AMD	235.33	679.27	1875
M1,0	64.40	80.40	39	M1,0	114.27	216.32	45	M1,0	200.17	266.85	162
M5,0	63.71	83.28	105	M5,0	109.15	195.51	98	M5,0	197.81	238.12	238
M5,20	62.56	79.67	151	M5,20	108.77	198.23	197	M5,20	195.53	244.45	433
M10,0	65.61	89.03	167	M10,0	109.90	199.72	167	M10,0	189.05	216.97	286
M10,20	60.72	76.38	257	M12,0	108.09	196.79	224	M12,0	190.60	225.05	319
M10,25	60.08	74.22	254	M6,0	107.19	186.65	111	M14,0	192.01	221.73	363
M12,30	67.33	104.69	307	M7,0	109.15	195.51	209	M16,0	186.89	213.54	397
M14,20	60.53	75.48	349	M8,0	107.63	191.29	139	M18,0	187.79	212.56	390
AMF	58.63	90.27	218	AMF	100.16	208.67	270	M20,0	189.22	219.71	429

Shaded entries show 'best' values; ^{A} $\times 10^6$, ^{B} $\times 10^9$, ^{C} time for ordering & symbolic factorisation (secs)

matrices with structure of the mixed model matrix, as encountered in REML analyses of large data sets in animal breeding. Implementation is simple, and reductions in computing time of 50% or more over minimum degree orderings are readily achieved.

REFERENCES

Amestoy, P. R., Duff, I. S. and Excellent, J.-Y. (1998) *Comp. Methods Appl. Mech. Eng.* **184**:501.
Amestoy, P. R., Duff, I. S., Koster, J. and Excellent, J.-Y. (2001) *SIAM J. Matrix Anal. Appl.* **23**:15.
Ducrocq, V. and Druet, T. (2003) In: *54th Annual Meeting*. Europ. Ass. Anim. Prod. Mimeo., 22pp.
George, A. and Liu, J. W.-H. (1981) *Computer Solution of Large Positive Definite Systems*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey 07632.
Karypis, G. and Kumar, V. (1998a) *SIAM J. Scient. Comp.* **20**:359.
Karypis, G. and Kumar, V. (1998b) METIS *A software package for partioning unstructured graphs, partitioning meshes, and computing fill-in reducing ordering of sparse matrices Version 4.0*. Department of Computer Science, University of Minnesota, Minneapolis, MN 55455. 44 pp.
Liu, J. W. H. (1985) *ACM Trans. Math. Soft.* **11**:141.
Meyer, K. (1998) DFREML version 3.0. CD-ROM of the Sixth World Congress on Genetics Applied to Livestock Production.
Meyer, K. (2005a) *J. Anim. Breed. Genet.* **122**:73.
Meyer, K. (2005b) *Proc. Ass. Advmt. Anim. Breed. Genet.* **16**:56.
Meyer, K., Johnston, D. J. and Graser, H.-U. (2004) *Austr. J. Agric. Res.* **55**:195.
Ng, E. G. and Raghavan, P. (1999) *SIAM J. Matrix Anal. Appl.* **20**:902.
Pellegrini, F., Roman, J. and Amestoy, P. (2000) *Concurrency: Practice and Experience* **12**:69.
Schulze, J. (2001) *BIT Numerical Mathematics* **41**:800.

ACKNOWLEDGEMENTS

This work was supported by the International Livestock Resources and Information Centre (ILRIC).