The Standards and Data Structures of Dense Linear Algebra Software

Fred Gustavson
IBM Research Center
Yorktown Height, NY, USA
Email: fg2@us.ibm.com

Jerzy Waśniewski (speaker)
Technical University of Denmark
Informatics & Mathematical Modeling
DK-2800 Lyngby, Denmark
Email: jw@imm.dtu.dk

The standard of the Dense Linear Algebra Software (DLAS) is LAPACK [4] and ScaLAPACK [5] libraries. They are supported by the level 1, 2 and 3 Basic Linear Algebra Subprograms [6, 7] (BLAS) and the Basic Linear Algebra Communication Subprograms [9] (BLACS). These software followed earlier standard software named LINPACK [8] and EISPACK [10]. There are also many other very good DLAS library packages.

The hardware of HPC (High Performance Computers) are ever changing, and new performance feature appear with each new platform. The above software can have its performance improved by modification that support these new features. A challenge for HPC is to incorporate these new features while maintaining validity and robustness of these existing standard software.

As an example of this process we demonstrate performance improvements of the DLAS for symmetric matrices.

LAPACK has two different kinds of algorithms for dense symmetric matrices [4]. The full storage data format and the packed storage data format. The full storage data format requires n^2 memory locations, and perform quickly. The LAPACK packed storage data format algorithms require minimal storage of size $n(n+1)/2$ but their speed is several times slower than of the LAPACK full storage data format algorithms.

Thus, the user's program should perform quickly and require minimum memory.

Algorithms using new packed storage data formats are very successful. They run with almost the same speed as the LAPACK full storage data format algorithms and only use $n(n+1)/2$ memory locations. Several factorization, solution and inversion algorithms for symmetric positive definite and indefinite matri-

ces has already been published [3, 2, 1, 12, 11], and their source programs are available.

A new data format, the Rectangular Full Packed (RFP) format will be presented. The RFP format can replace both LAPACK formats for symmetric and triangular matrices, the full and packed formats. The RFP data format algorithms require minimal storage of size $n(n+1)/2$ but their speed is almost the same speed as the LAPACK full storage data format algorithms.

The standard two dimensional arrays of Fortran and C that are used to store triangular and symmetric matrices waste half the storage space but provide high performance via the use of level 3 BLAS. Packed format arrays fully utilize storage but provide low performance as there are no level 3 packed BLAS. We combine the good features of packed and full storage using RFP format to obtain high performance using L3 (Level 3) BLAS as RFP is totally full format. Also RFP format requires exactly the same minimal storage as packed storage. Each full and/or packed symmetric/triangular routine becomes a single new RFP routine. The idea of the RFP format will also introductory be presented at PARA'06 in Umeå, Sweden.

Performance results of factorization, solution and inversion subroutines between LAPACK full and packed storages, and algorithms based on the RFP data storage on several different computers will be presented. The new data format RFP and algorithm will be explained.

References

[1] B.S. Andersen, J. Gunnels, F.G. Gustavson, J.K. Reid, J. Waśniewski. A fully portable High Performance Minimal Storage Cholesky Algorithm. ACM Trans. on Math. Software 31, 2005, 201-227.

[2] Andersen, B., Gunnels, J., Gustavson, F., and Waśniewski, J., Recursive Formulation of the Inversion of symmetric positive definite Matrices in Packed Storage Data Format. In J. Fagerholm, J. Haataja, J. Järvinen, M. Lyly, and P. R. V. Savolainen Eds., Proceedings of the 6th International Conference, PARA 2002, Applied Parallel Computing, Number 2367 in Lecture Notes in Computer Science. Espoo, Finland, June 2002, pp. 287–296. Springer.

[3] B. S. Andersen, F. G. Gustavson, J. Waśniewski. A recursive formulation of Cholesky factorization of a matrix in packed storage. *TOMS*, Vol. 27, No. 2, June. 2001, pp. 214-244.

[4] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide.* Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.

[5] L.S. Blackford, J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker,

and R.C. Whaley. "ScaLAPACK Users' Guide". Society for Industrial and Applied Mathematics, 1997, Philadelphia, PA.

[6] L.S. Blackford, J. Demmel, J. Dongarra, I. Duff, S. Hammarling, G. Henry, M. Heroux, L. Kaufman, A. Lumsdaine, A. Petitet, R. Pozo, K. Remington, R.C. Whaley, An Updated Set of Basic Linear Algebra Subprograms (BLAS), ACM Trans. Math. Soft., 28-2 (2002), pp. 135–151.

[7] J. Dongarra, Basic Linear Algebra Subprograms Technical Forum Standard, International Journal of High Performance Applications and Supercomputing, 16(1) (2002), pp. 1–111, and International Journal of High Performance Applications and Supercomputing, 16(2) (2002), pp. 115–199.

[8] J.J. Dongarra, J.R. Bunch, C.B. Moler, and G.W. Stewart, LINPACK Users' Guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1979.

[9] J. Dongarra and R.C. Whaley, A user's guide to the BLACS v1.1, Computer Science Dept. Technical Report CS-95-281, University of Tennessee, Knoxville, TN, 1995. (Also LAPACK Working Note #94).

[10] B.S. Garbow, J.M. Boyle, J.J. Dongarra, and C.B. Moler, Matrix Eigensystem Routines - EISPACK Guide Extension, vol. 51 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1977.

[11] F. Gustavson, A. Karaivanov, M. Marinova, J. Waśniewski, and P. Yalamov. "A Fast Minimal Storage Symmetric Indefinite Solver". In Para'2000 Conference Proceedings, Lecture Notes in Computer Science Nr. 1947, Bergen, Norway, 2000.

[12] F.G. Gustavson, J.K. Reid, and J. Waśniewski. Algorithm xxx: Fortran 95 Subroutines for Cholesky Factorization and Solution in Blocked Packed Hybrid Format. Already accepted and submitted after referee to Trans. of Math. Software of ACM. March 2006.