

Recursive Blocked Algorithms for Dense Matrix Computations

Erik Elmroth^a, Fred Gustavson^b, Isak Jonsson^a and Bo Kågström^a

^aDepartment of Computing Science and HPC2N
Umeå University, SE-901 87 Umeå, SWEDEN
{elmroth, isak, bokg}@cs.umu.se

^bIBM T. J. Watson Research Center
P.O. Box 218, Yorktown Heights, NY 10598, U.S.A.
gustav@watson.ibm.com

Today block-based algorithms are the standard in state-of-the-art software for dense linear algebra computations. These algorithms perform most of their computations in calls to high-performance level 3 BLAS. The level 3 BLAS obtain most of their performance by data rearrangements and the use of high-performance atomic kernel routines. The current state-of-the-art BLAS are designed to exploit the architecture design while maintaining the functionality of the BLAS and thereby guarantee high performance and portability of dense linear algebra codes [8, 9, 5]. The level 3 factorization algorithms typified by LAPACK make repeated calls to the level 3 BLAS with matrix operands equal to submatrices of fixed size. This results in multiple data copying on operands that are related. How can this be challenged? An answer is to change the data structure of the BLAS input matrices to reflect this relationship, thereby removing any data copying from the BLAS. The obvious choice is to store matrices as blocks, i.e., submatrices of block-partioned matrices, instead of the standard column-major (Fortran) or row-major (C) orderings.

Recursion is a key concept for matching an algorithm and its data structure. A recursive algorithm leads to automatic blocking which is variable and “squarish” [3]. This layered and variable blocking allow for good data locality, which makes it possible to approach peak performance on today’s memory tiered processors.

In this contribution, we give an overview of our research in recursive blocked algorithms and data formats for dense matrix computations, including level 3 BLAS [4, 5], dense matrix factorizations for full storage [3, 1, 2] and packed storage [6], and triangular Sylvester-type matrix equations [7]. One important application of triangular matrix equations is condition estimation and error bounds for different eigenspace problems [10, 11, 12].

This research was conducted using the resources of the High Performance Computing Center North (HPC2N) and UNI-C.

References

- [1] E. Elmroth and F. Gustavson. New Serial and Parallel Recursive *QR* Factorization Algorithms for SMP Systems, In Kågström et al. (eds), *Applied Parallel Computing, PARA’98*, Lecture Notes in Computer Science, Vol. 1541, pp 120–128, Springer-Verlag, 1998.
- [2] E. Elmroth and F. Gustavson. Applying Recursion to serial and Parallel QR Factorization Leads to Better Performance. Submitted to *IBM Journal of Research and Development*, 1999.
- [3] F. Gustavson. Recursion leads to automatic variable blocking for dense linear algebra. *IBM J. Res. Develop*, 41(6):737–755, November 1997.

- [4] F. Gustavson, A. Henriksson, I. Jonsson, B. Kågström and P. Ling. Recursive Blocked Data Formats and BLAS's for Dense Linear Algebra Algorithms. In Kågström et al. (eds), *Applied Parallel Computing, PARA '98*, Lecture Notes in Computer Science, Vol. 1541, pp 195–206, Springer-Verlag, 1998.
- [5] F. Gustavson, A. Henriksson, I. Jonsson, B. Kågström and P. Ling. Superscalar GEMM-based Level 3 BLAS – The On-going Evolution of a Portable and High-Performance Library. In Kågström et al. (eds), *Applied Parallel Computing, PARA '98*, Lecture Notes in Computer Science, Vol. 1541, pp 207-215, Springer-Verlag, 1998.
- [6] F. Gustavson and I. Jonsson. Recursive Level 3 Cholesky Factorization Implementation for Recursive Packed Data. Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden. *In preparation*, 1999.
- [7] I. Jonsson and B. Kågström. Recursive Blocked Algorithms for Solving Triangular Sylvester-Type Matrix Equations. Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden. *In preparation*, 1999.
- [8] B. Kågström, P. Ling, and C. Van Loan. GEMM-based level 3 BLAS: High-performance model implementations and performance evaluation benchmark. *ACM Trans. Math. Software*, 24(3):268–302, 1998.
- [9] B. Kågström, P. Ling, and C. Van Loan. GEMM-based level 3 BLAS: Portability and optimization issues. *ACM Trans. Math. Software*, 24(3):303–316, 1998.
- [10] B. Kågström and P. Poromaa. Distributed and Shared Memory Block Algorithms for the Triangular Sylvester Equation with sep^{-1} Estimator. *SIAM Journal on Matrix Analysis and Application*, 13(1):90–101, January 1992.
- [11] B. Kågström and P. Poromaa. LAPACK-Style Algorithms and Software for Solving the Generalized Sylvester Equation and Estimating the Separation between Regular Matrix Pairs. *ACM Trans. Math. Software*, 22(1):78–103, March 1996.
- [12] P. Poromaa. Parallel Algorithms for Triangular Sylvester Equations: Design, Scheduling and Scalability Issues. In Kågström et al. (eds), *Applied Parallel Computing, PARA '98*, Lecture Notes in Computer Science, Vol. 1541, pp 438–446, Springer-Verlag, 1998.