جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

Accelerating the Dense Singular Value Decomposition Using High Performance Polar Decomposition Implementation

Dalal Sukkari, David Keyes and Hatem Ltaief
Extreme Computing Research Center
King Abdullah University of Science and Technology

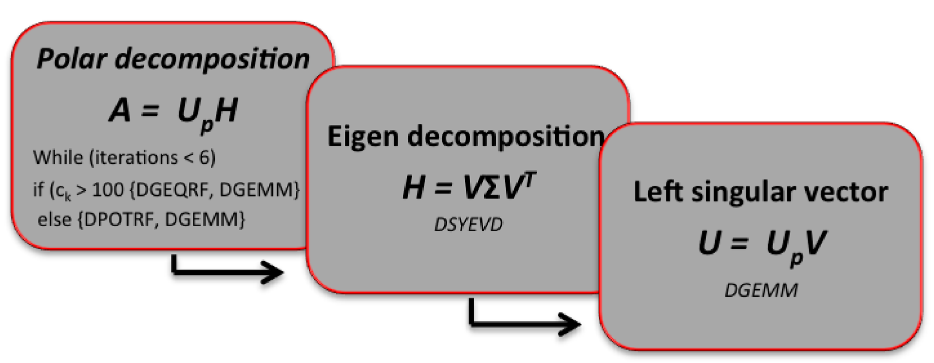
Motivations

The Singular Value Decomposition (SVD) is a crucial matrix operations. It is used as a basic block in various numerical algorithms and applications such as, solving least squares problem, computing the principal component analysis in statistics, data clustering algorithms, find a low rank matrix approximation, etc. Modern graphics processing units (GPUs) are powerful hardware accelerators, which can sometimes revive arithmetically expensive numerical algorithms previously not considered by bringing them up to speed and making them competitive against equivalent implementations. This is the case for an SVD solver (QDWH-SVD) based on the polar decomposition, which outperforms existing state-of-the-art SVD solvers, although it exhibits at a glance a prohibitive number of extra floating-point operations.

QDWH-SVD

Problem Definition: for a $m \times n$ real matrix A, the singular value decomposition is $A = U\Sigma V^\top$, where Σ is the matrix of all the singular values, U, V are the matrices of corresponding singular vectors.

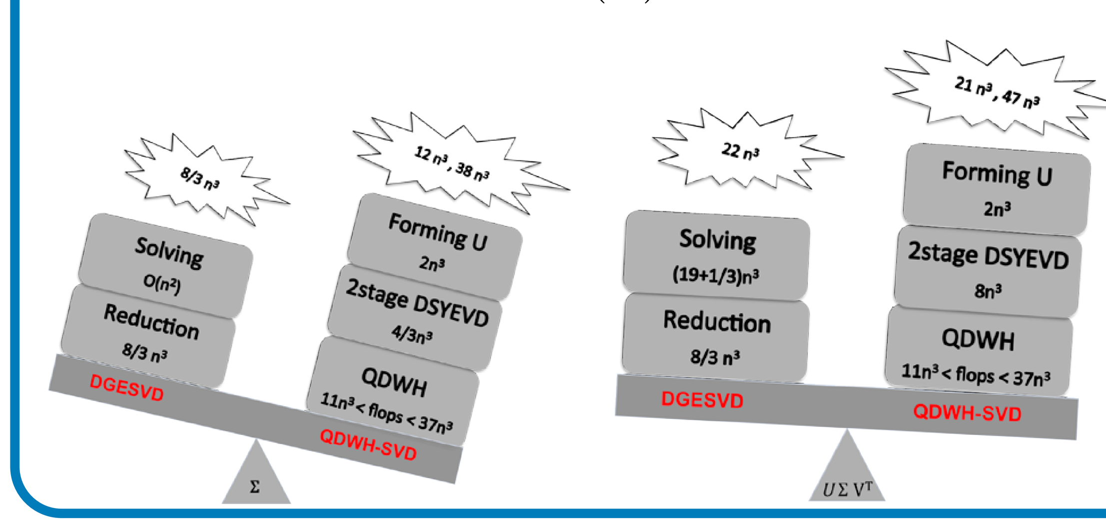
QDWH-SVD [1]:



- QDWH-SVD is a backward stable algorithm.
- The main part of QDWH-SVD is the polar decomposition (QDWH).

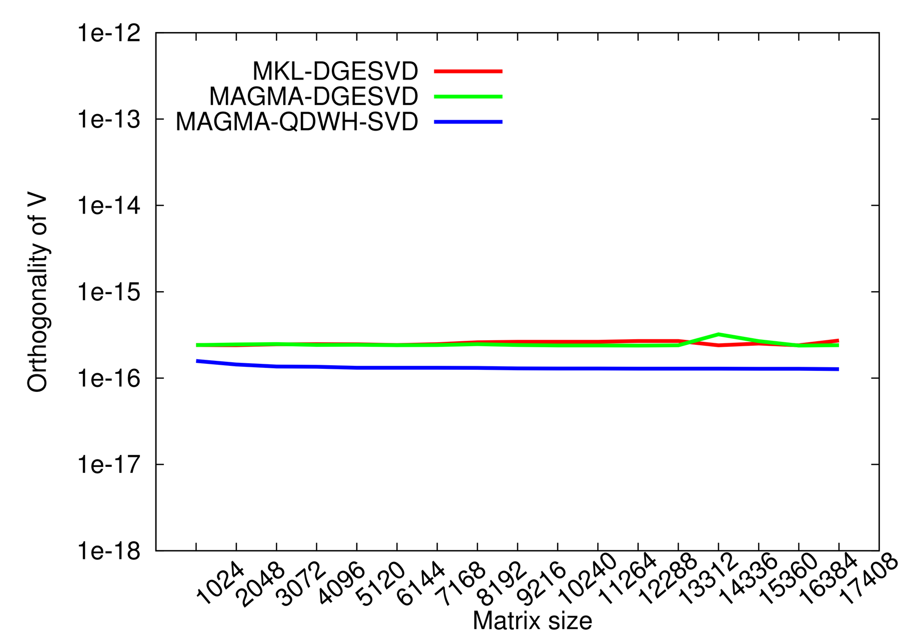
QDWH:

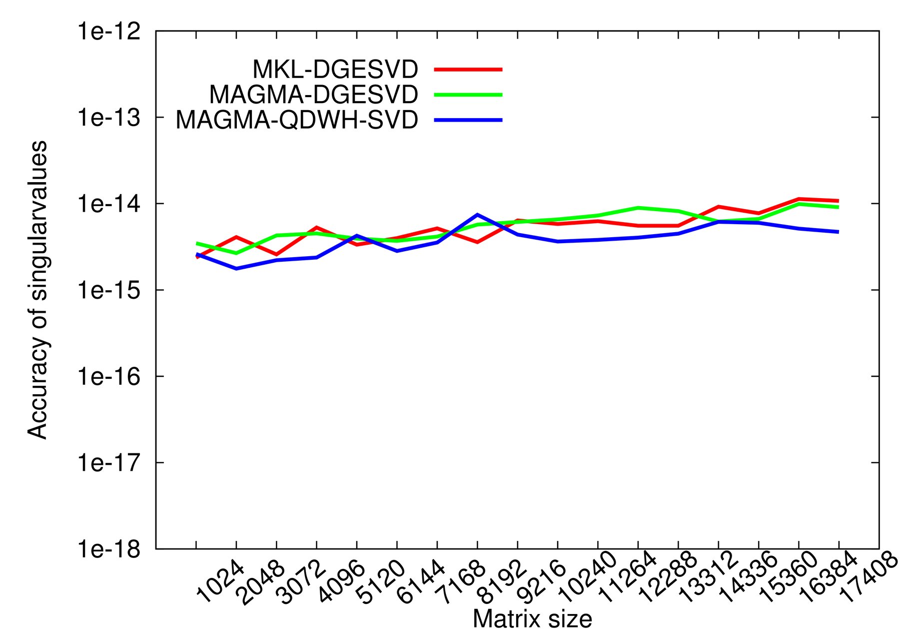
- Building blocks are matrix multiplication, QR and Cholesky factorizations.
- QDWH converges at most after 6 iterations.
- The flop count of QDWH depend on the matrix condition number $\kappa_2(A)$.

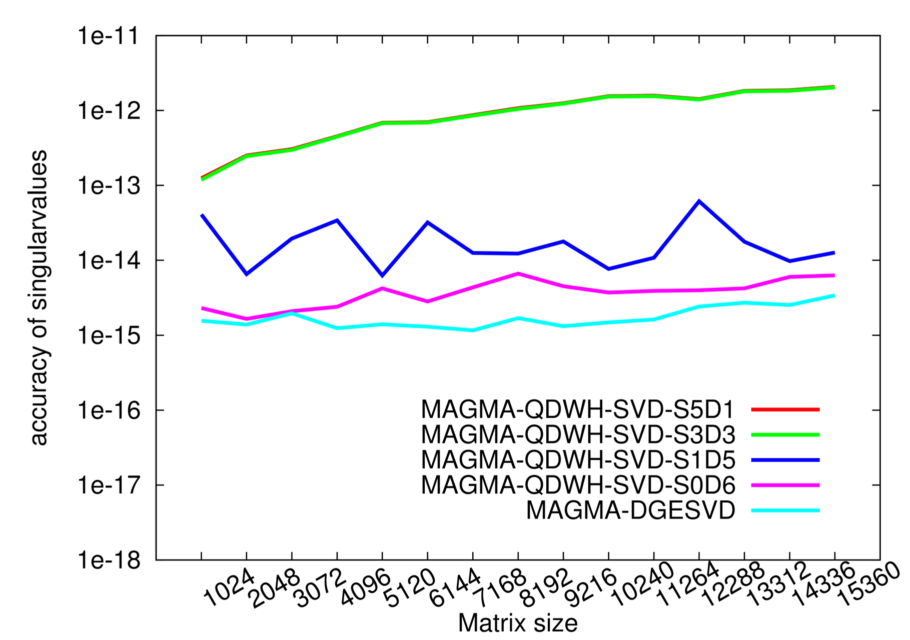


Accuracy Results

All the results are for ill-conditioned matrices generated using DLATMS from LAPACK with $\kappa_2(A) = 1/\epsilon$.







Environment Settings

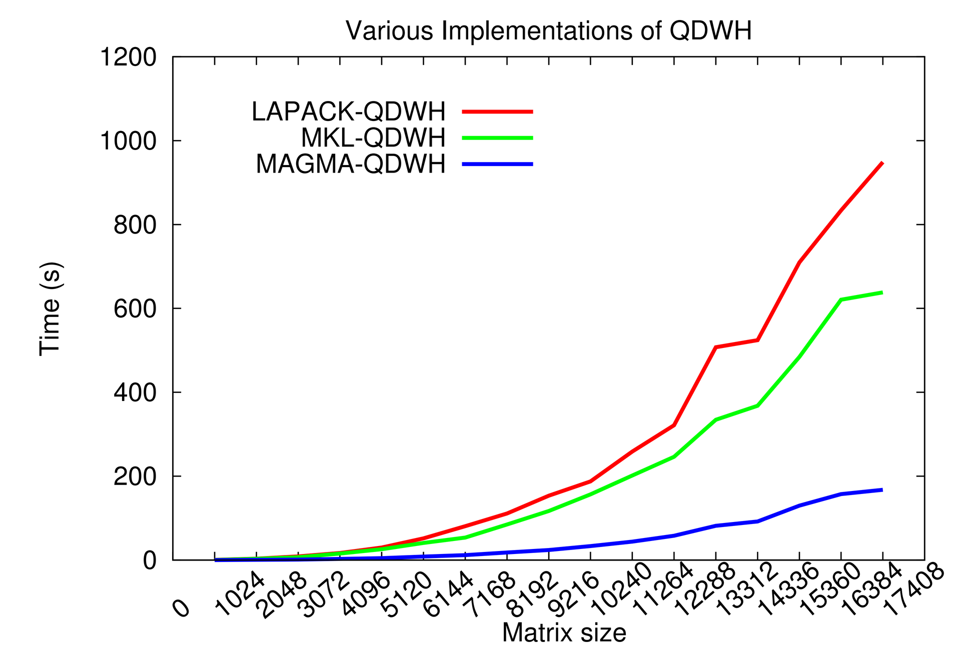
Software:

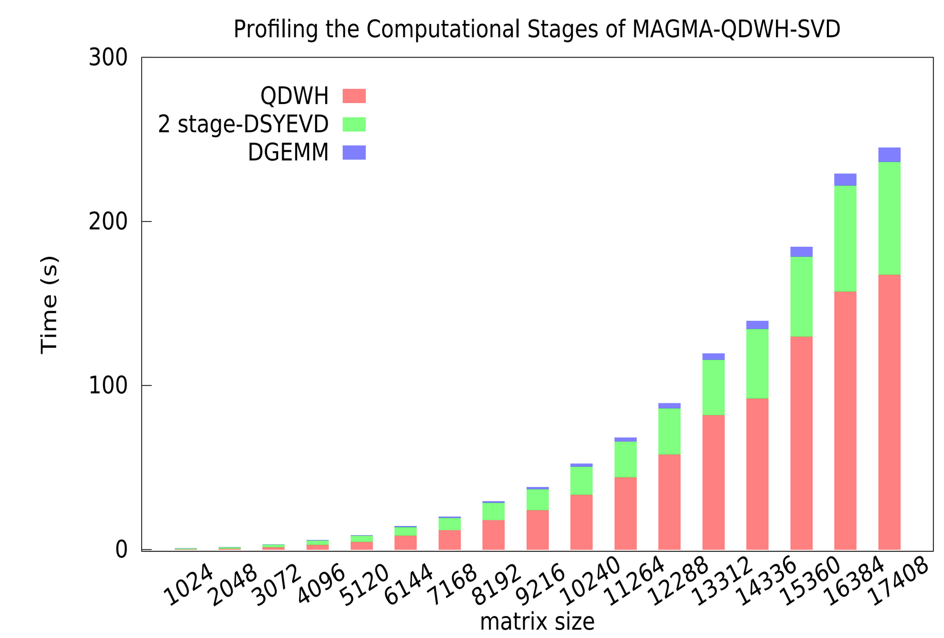
- Intel Compiler Suite v13.0.1.
- NVIDIA CUDA v6.0.
- MAGMA v1.4.1 [2] (compiled w/ -mkl=parallel).
- KBLAS v1.0 [3].

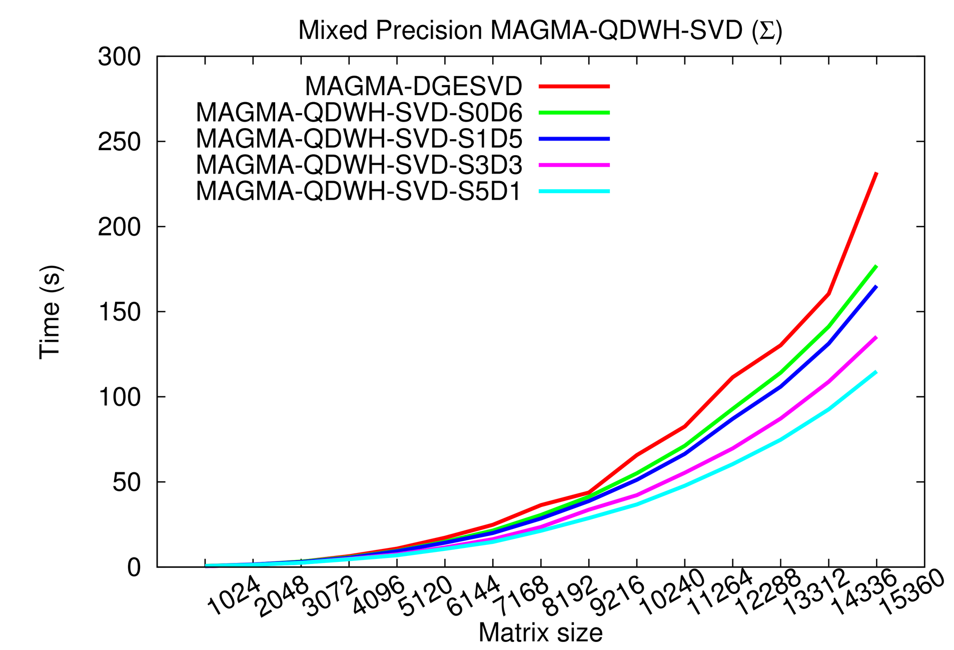
Hardware:

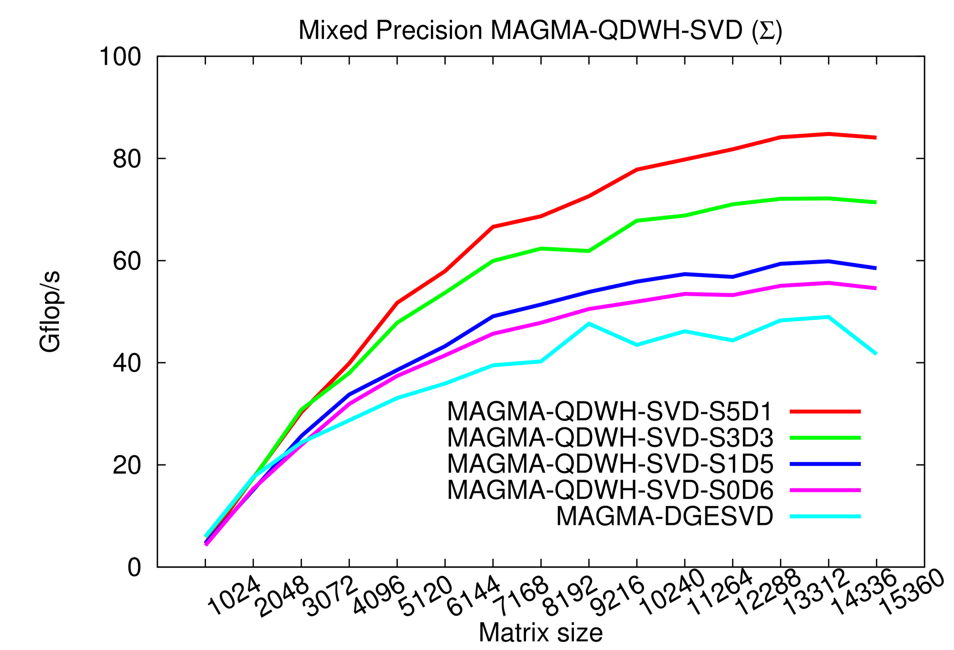
- Intel(R) Xeon(R) CPU E5-2680 v2.
- Dual-socket 10-core Ivy Bridge (20 cores total).
- 256 GB of DDR3 main memory, 25 MB Cache, 2.8 GHz, 8.00 GT/s Intel(R) QPI.
- Three Tesla K40c (ECC off), 745 MHz clock, 12 GB.

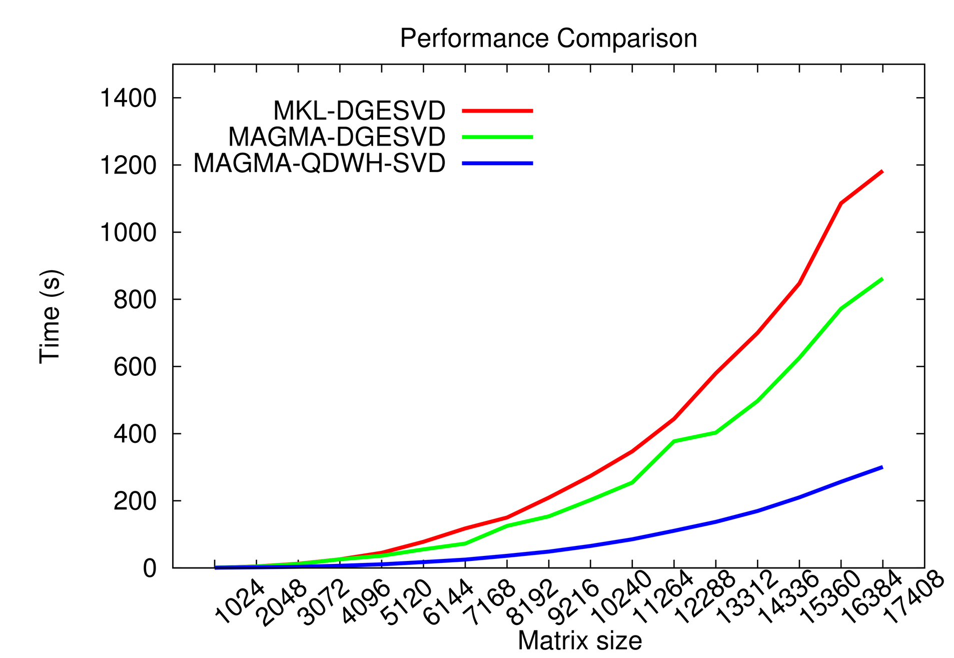
Performance Results

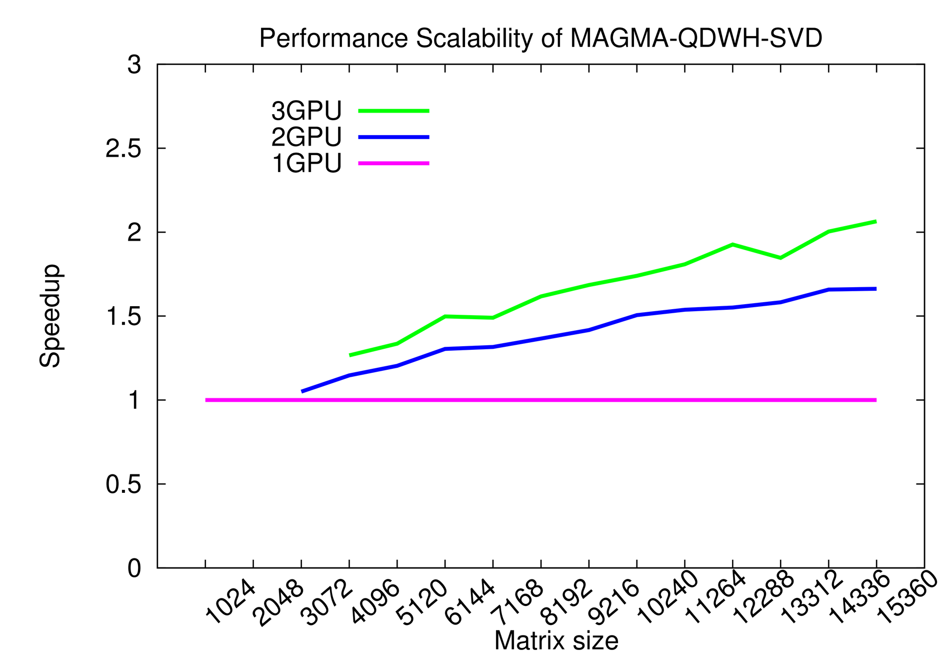












Future Research Directions

- Use the highly optimized Hierarchical QR (University of Colorado Denver) and 2.5D Matrix-Matrix multiplications (University of California Berkeley).
- Replace current block algorithms (coarse-grained) used in the QDWH-SVD framework by tile algorithms (fine-grained).
- Implement QDWH-SVD on distributed memory systems (w/ the divide-and-conquer QDWH-EIG instead).

References

[1] Y. Nakatsukasa and N. Higham. Stable and Efficient Spectral Divide and Conquer Algorithms for the Symmetric Eigenvalue Decomposition and the SVD. *SIAM Journal on Scientific Computing*, 35(3):A1325–A1349, 2013.

[2] E. Agullo, J. Demmel, J. Dongarra, B. Hadri, J. Kurzak, J. Langou, H. Ltaief, P. Luszczek, and S. Tomov. Numerical Linear Algebra on Emerging Architectures: The PLASMA and MAGMA projects. *Journal of Physics: Conference Series*, 180(1):012037, 2009.

[3] A. Abdelfattah, D. Keyes, and H. Ltaief. KBLAS: An Optimized Library for Dense Matrix-Vector Multiplication on GPU Accelerators. *Submitted to TOMS*, 2014.