Numerical Computation

Triple and Quadruple Precision BLAS Subroutines on GPUs

Background

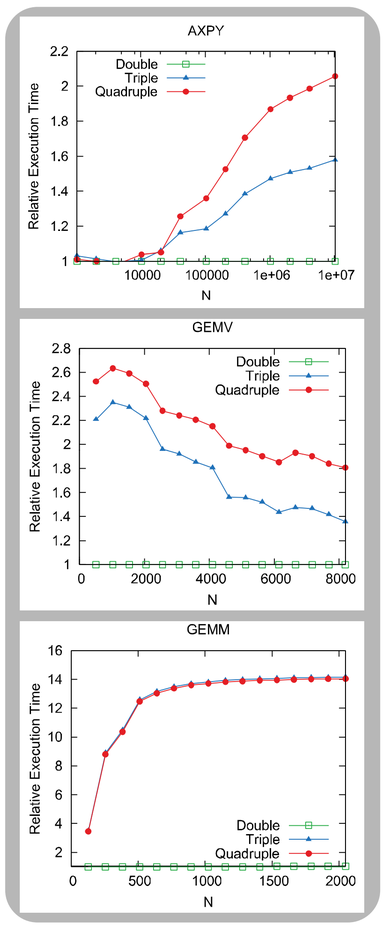
Floating - point operations have round - off errors. These errors may become a critical issue for some applications. Especially in large - scale computing, an accumulation of round – off errors may become a more serious problem. Double precision accuracy may be insufficient in some cases, and there is a demand for higher precision operations.

Overview

We have implemented triple and quadruple precision Basic Linear Algebra Subprograms (BLAS) subroutines, AXPY (y = αx+y), GEMV (y = αAx+βy) and GEMM (C = αAB+βC) on GPUs. For quadruple precision, we used Double – Double (DD) type quadruple precision operations (11bits exponent & 104bits significand). On the other hand, we propose Double+Single (D+S) type triple precision floating – point format (8bites exponent & 75 bits significand) and triple precision operations using DD – operations internally.

Performance

Some level - 1 and 2 BLAS subroutines are memory - bound on the Tesla M2090, not only in single and double, but also triple and quadruple precision: the execution time of triple and quadruple precision subroutines is close to only 1.5x and 2.0x of that of double precision. For memory - bound operations where double precision is insufficient but quadruple precision is not needed, triple precision operations should be used.



Auto - Tuning of SpMV for CRS format on GPUs

Background

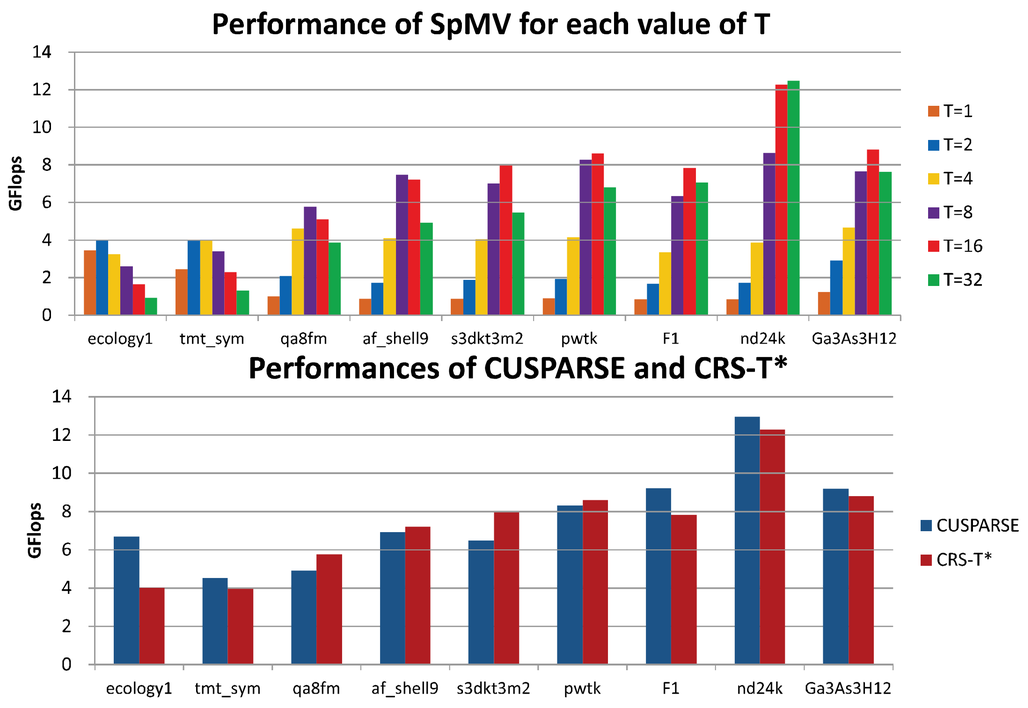
Performance of sparse matrix - vector multiplication (SpMV) on GPUs is highly dependent on the structure of the sparse matrix used in the computation, the computing environment, and the selection of certain parameters.

Overview

We show that the performance achieved using kernel SpMV on GPUs for the compressed row storage (CRS) format depends greatly on optimal selection of a parameter T that is a number of threads to compute an output vector element, and we propose an efficient algorithm for the automatic selection of the optimal parameter.

Performance

CRS - T* that is kernel SpMV for the CRS format using automatic parameter selection achieves up to approximately 26% improvement over NVIDIA's CUSPARSE library.

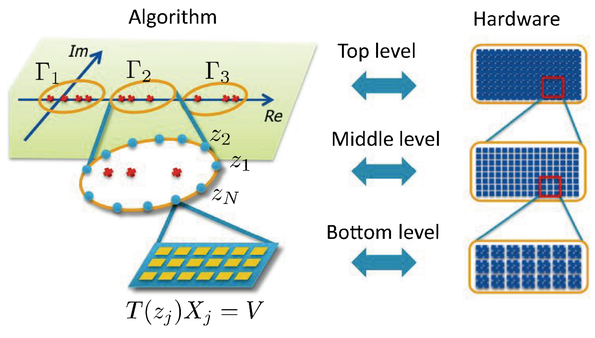


Development of Parallel Sparse Eigensolver Package: z - Pares

The aim of this research project is to develop numerical software for large-scale eigenvalue problems in post-petascale computing environment. An eigensolver based on contour integral (the SS method) has been proposed by Sakurai and Sugiura [3]. This method has a hierarchical structure and is suitable for massively parallel supercomputers [2]. Moreover, the SS method can be applicable for nonlinear eigenvalue problem [1]. Block Krylov method [4] improve the performance of the method. We are developing software on both Trilinos and PETSc. MATLAB version is available at http://zpares.cs.tsukuba.ac.jp/

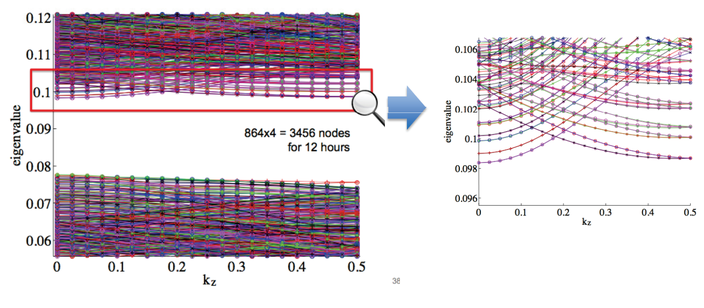
Hierarchical Parallel Structure

Hardware is grouped according to a hierarchical structure of the algorithm.



Numerical Example on the K Computer

Application for band calculation with real space density functional theory (RSDFT) [2].



Band structure of silicon nanowire of 9,924 atoms. (matrix size = 8,719,488, Number of cores = 6,144)

*The results are tentative since they are obtained by early access to the K computer.

Reference

[1] J. Asakura, T. Sakurai, H. Tadano, T. Ikegami and K. Kimura, A numerical method for nonlinear eigenvalue problems using contour integrals, JSIAM Letters, 1 (2009) 52-55.

[2] Y. Futamura, T. Sakurai, S. Furuya and J.-I. Iwata, Efficient algorithm for linear systems arising in solutions of eigenproblems and its application to electronic-structure calculations, Proc. 10th International Meeting on High-Performance Computing for Computational Science (VECPAR 2012) (accepted).

[3] T. Sakurai and H. Sugiura, A projection method for generalized eigenvalue problems, J. Comput. Appl. Math., 159 (2003) 119-128.

[4] H. Tadano, T. Sakurai and Y. Kuramashi, Block BiCGGR: A new block Krylov subspace method for computing high accuracy solutions, JSIAM Letters, 1 (2009) 44-47.

Acknowledgment

This research was partially supported by the Core Research for Evolutional Science and Technology (CREST) Project of JST and the Grant-in-Aid for Scientific Re- search of Ministry of Education, Culture, Sports, Science and Technology, Japan, Grant number: 23105702.