Parallel Numerics '05, 7 M. Vajteršic, R. Trobec, P. Zinterhof, A. Uhl (Eds.)
Chapter 1: Abstracts of Invited Talks ISBN 961-6303-67-8

New Jacobi–Type Algorithm for Computing the Singular Value Decomposition

Zlatko Drmač *

Department of Mathematics
University of Zagreb
Bijenička 30, 10000 Zagreb, Croatia

Many problems in science and engineering are stated as matrix eigenvalue and singular value decomposition problems. In this talk, we describe high–performance computation of the matrix singular value decomposition and show how to use it to solve a family of generalized eigenvalue and singular value problems. Our target architectures are modern high performance single–processor machines, workstation networks, as well as modern supercomputers. Wanted is a unified approach that enables efficient construction of high–performance code for a variety of modern computers. The kernel procedures are from the BLAS, LAPACK, BLACS, PBLAS and ScaLAPACK libraries.

The problem of tradeoff between accuracy and efficiency is one of the core problems in matrix computations software development. More accurate algorithms are usually less efficient than less accurate but much faster algorithms. In the case of SVD, it is known that the one–sided Jacobi algorithm is more accurate but much slower than the methods based on bidiagonalization (such as the QR and the divide and conquer method). On the other hand, the one–sided Jacobi algorithm is perfect for parallelization and it has been studied by many researches. Many parallel strategies and other techniques are developed to improve the performance of the Jacobi algorithm in parallel environment.

The focus of our talk are new results regarding the convergence behavior of a modified Jacobi SVD algorithm. In our opinion, this new implementation of the Jacobi algorithm, with swift numerical convergence, will require new techniques for efficient parallelization.

*Corresponding author. E-mail: drmac@math.hr

We will present a serial implementation of the new Jacobi SVD algorithm and show its superior numerical properties (as compared to bidiagonalization based methods) as well as competitive efficiency. The new algorithm will be included in the next version of the LAPACK library and right now we have started thinking on its ScaLAPACK–style implementation. We will outline many interesting problems related to the parallelization of the new algorithm.