Parallel Computation of the Generalized Singular Value Decomposition*

Zhaojun Bai
Courant Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, NY 10012, USA

Abstract

The generalized singular value decomposition (GSVD) is the simultaneous reduction of any two matrices having the same number of columns to diagonal forms by premultiplying by two different orthogonal matrices and postmultiplying by the same nonsingular matrix. The recent advent of real time signal processing and other problems have given impetus to the development of the parallel computation of the GSVD [3,7]. Brent et al. [2] have developed a parallel algorithm on systolic array. However, their approach requires the use of different configurations of systolic arrays for singular value decomposition (SVD), QR decomposition and matrix-matrix multiplication. In addition to some difficulties in numerical treatment, how to avoid costly interfacing between these arrays and devise a single array that is capable of performing all these basic matrix computations is still an open problem.

Recently, Luk [5] proposed a parallel implementation of Paige's Jacobi-like algorithm[6]. But in his work, only the simplest case of

*Prepared for an Interdisciplinary Meeting on Numerical Linear Algebra, Digital Signal Processing, and Parallel Algorithms, August 1988. This work was supported in part by the U.S. Air Force Office of Scientific Research under Grant AFOSR-82-0078.
computing the GSVD of two $n \times n$ triangular matrices, one of which is nonsingular, is considered. The scheme cannot treat the general case.

Our parallel direct GSVD algorithm will treat the general case. The algorithm falls into two parts: first, the input matrices are preprocessed in parallel by computing their upper trapezoidal forms, for which we develop the parallel QR decomposition with column pivoting. And second, the GSVD of two upper trapezoidal matrices is computed in parallel using generalized Jacobi-Kogbetliantz-iteration scheme. The 2-by-2 GSVD is developed to be used in the forward and backward sweeps to keep the trapezoidal structure. We also consider in detail how to organize the computation, how to partition the matrix and how to map these pieces of data and computations onto the network of processors. The whole algorithm is designed to be implemented efficiently on distributed-memory parallel computer architectures. The time is $O(n^2)$ for parallel-preprocessing, and $O(n^2/p)$ for the GSVD of two upper trapezoidal matrices, where $p$ is the dimension of the triangular array of processors. In fact, the computation of the GSVD of two upper trapezoidal matrices can also be implemented directly on a triangular systolic array. How to efficiently compute the QR decomposition with column pivoting on systolic array is still unknown, although there are many papers concerning the QR decomposition without column pivoting on systolic array, e.g. see [4].

References


