Fast Parallel Orthogonalization

Dexter Kozen
Computer Science Department
Cornell University
Ithaca, New York 14853

Given a linearly independent set of vectors $x_1, \ldots, x_n \in \mathbf{C}^d$, the Gram-Schmidt orthogonalization procedure produces a new set $y_1, \ldots, y_n$ such that y_i is contained in the linear span of $x_1, \ldots, x_i$, $1 \le i \le n$, and $y_i^T \bar{y}_j = 0$, $i \ne j$. The y_i are produced sequentially, using the formula

$$y_i = x_i - \sum_{j=1}^{i-1} \frac{x_i^T \bar{y}_j}{y_j^T \bar{y}_j} \cdot y_j$$

There is a simple NC algorithm for producing the y_i. Let A be a square matrix with the property that all principal minors are nonsingular. Let

$$\left| A_{j_1, \ldots, j_m}^{i_1, \ldots, i_m} \right|$$

denote the determinant of the submatrix of A consisting of rows $i_1, \ldots, i_m$ and columns $j_1, j_2, \ldots, j_m$. Then $A = LU$, where

$$L_{ij} = \frac{\left| A_{1,2,\ldots,j-1,j}^{1,2,\ldots,j-1,i} \right|}{\left| A_{1,2,\ldots,j}^{1,2,\ldots,j} \right|}, \qquad U_{ij} = \frac{\left| A_{1,2,\ldots,i-1,j}^{1,2,\ldots,i-1,i} \right|}{\left| A_{1,2,\ldots,i-1}^{1,2,\ldots,i-1} \right|}.$$

Note that L is lower triangular and U is upper triangular. L and U can be computed in NC using Csanky's algorithm [2].

Now let P be the $d \times n$ matrix whose columns are the given vectors $x_1, \ldots, x_n$. By results of [1,3], we can assume without loss of generality that the x_i are linearly independent. Then all principal minors of $P^T \bar{P}$ are nonsingular. Using the LU algorithm above, compute

$$P^T \bar{P} = LU, \qquad Q = P(L^{-1})^T.$$

Let $y_1, \ldots, y_n$ be the columns of Q. Then y_i is contained in the linear span of $x_1, \ldots, x_i$ since $(L^{-1})^T$ is upper triangular; and the y_i are orthogonal, since

$$Q^T \bar{Q} = L^{-1} P^T \bar{P} (\bar{L}^{-1})^T = L^{-1} L U (\bar{L}^{-1})^T = U (\bar{L}^{-1})^T$$

is upper triangular and Hermitian, thus diagonal.

[1] Borodin, A., J. von zur Gathen, and J. Hopcroft, "Fast Parallel Matrix and GCD Computations," *Information and Control* 52:3 (1982), 241-256.
[2] Csanky, L., "Fast parallel matrix inversion algorithms," *SIAM J. Comput.* 5 (1976), 618-623.
[3] Ibarra, O., S. Moran, and L.E. Rosier, "A note on the parallel complexity of computing the rank of order n matrices," *Info. Proc. Letters* 11 (1980), 162.