A New Communication-Efficient Distributed Fourier Transform Algorithm

Peter McCorquodale*
Sivan Toledo†
MIT Laboratory for Computer Science

The discrete Fourier transform $y = \mathrm{DFT}(x)$ arises frequently in scientific and engineering applications. It is defined as the linear transformation

$$y = Fx \tag{1}$$

where x and y are real- or complex-valued vectors of length n, and F, the Fourier matrix of order n, is given by

$$F_{jk} = \frac{1}{\sqrt{n}} e^{\frac{2\pi ijk}{n}}; \qquad 0 \leq j, k \leq n-1. \tag{2}$$

The well-known Fast Fourier Transform (FFT) algorithm [CT65] takes advantage of the group structure of this matrix to compute the DFT using $O(n \log n)$ sequential floating-point operations.

In the parallel distributed-memory version of the DFT, there are p processors (labelled $0, 1, \ldots, p-1$), each containing $\frac{n}{p}$ components of x with consecutive indices. Upon termination of the algorithm, each processor should contain $\frac{n}{p}$ components of y with consecutive indices. Distributed versions of the FFT algorithm require $O(\frac{n}{p} \log n)$ flops per processor, and a total of at least $n \log p$ scalars communicated between processors [JJK92], [VL92]. In this abstract, we describe an algorithm that reduces this communication to $n + O(p^2 \log \frac{n}{p})$ scalars and uses $O(\frac{n^2}{p^2})$ flops per processor. This algorithm may be preferable to the FFT if arithmetic is cheap and communication is relatively expensive.

The new algorithm uses a block decomposition of Equation (1), as shown in Figure 1. We may take a singular value decomposition of the order-$\frac{n}{p}$ matrix A [GVL89]:

$$A = USV^*, \tag{3}$$

where U and V are unitary matrices, and S is a diagonal matrix with nonnegative diagonal entries, called the singular values of A. The second author observed empirically that when $p = 2$, nearly half of the singular values are 0, and nearly half are 1 [Tol95]. The first author proved that for arbitrary p, a fraction of almost $\frac{1}{p}$ of the singular values are close to 1, a fraction of almost $1 - \frac{1}{p}$ are close to 0, and the fraction of singular values far from 0 and 1 is asymptotically small [McC95].

From Figure 1 and Equation (1), we have

$$y_s = \sum_{r=0}^{p-1} \alpha^{rs} D^r U z_{sr} \tag{4}$$

where

$$z_{sr} = SV^* D^s x_r \tag{5}$$

is a vector of $\frac{n}{p}$ scalars to be sent from processor r to processor s. Since S is diagonal

*Peter McCorquodale is advised by Alan Edelman. Address: Room 311, 545 Technology Square, Cambridge, MA 02139. E-mail: `petermc@theory.lcs.mit.edu`

†Sivan Toledo is advised by Charles Leiserson and is supported in part by ARPA under grant N00014-94-1-0985. Address: Room 243, 545 Technology Square, Cambridge MA 02139. E-mail: `sivan@mit.edu`

$$\begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{p-1} \end{pmatrix} = \begin{pmatrix} A & DA & \cdots & D^{p-1}A \\ AD & \alpha DAD & \cdots & \alpha^{p-1}D^{p-1}AD \\ \vdots & \vdots & & \vdots \\ AD^{p-1} & \alpha^{p-1}DAD^{p-1} & \cdots & \alpha^{(p-1)^2}D^{p-1}AD^{p-1} \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{p-1} \end{pmatrix}$$

Figure 1: Block decomposition of Fourier matrix multiplication, $y = Fx$.
D is a fixed unitary diagonal matrix, and α is a complex constant of magnitude 1.

with only a little more than $\frac{n}{p^2}$ nonzero entries (within machine precision), only about $\frac{n}{p^2}$ components of z_{sr} need to be communicated.

For p up to 1024, it was found that $\frac{n}{p^2} + 2 \log \frac{n}{p}$ scalars sent between each pair of processors suffice to approximate the DFT to within machine precision using 64-bit complex arithmetic. Note also that all communication is done at the same stage in the algorithm, and takes the form of all-to-all personalized communication.

The floating point operations involved are the matrix multiplications that each processor performs to calculate the vectors z_{rs} and then y_s. Each of these two matrix multiplications can be done using $O(\frac{n^2}{p^2})$ flops.

References

[CT65] J. W. Cooley and J. W. Tukey. An Algorithm for the Machine Calculation of Complex Fourier Series. *Math. Comp.*, 19:297–301 (1965).

[GVL89] Gene H. Golub and Charles Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, 1989.

[JJK92] S, Lennart Johnsson, Michel Jacquemin, and Robert L. Krawitz. Communication Efficient Multi-processor FFT. *J. Comp. Phys.*, 102:381–397 (1992).

[McC95] Peter McCorquodale. Singular value analysis of a new distributed Fourier transform algorithm. Manuscript, 1995.

[Tol95] Sivan Toledo. On the Communication Complexity of the Discrete Fourier Transform. Manuscript, 1995.

[VL92] Charles Van Loan. *Computational frameworks for the fast Fourier transform.* SIAM, Philadelphia, 1992.