ROUNDOFF ERROR ANALYSIS OF THE FAST FOURIER TRANSFORM

BY

GEORGE U. RAMOS

STAN-CS-70-146
FEBRUARY 1970

COMPUTER SCIENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

ROUNDOFF ERROR ANALYSIS

OF THE

FAST FOURIER TRANSFORM

by

George U. Ramos

The preparation of this report was sponsored by the Office of Naval Research under grant number N0013-67-A-0112-0029, the National. Science Foundation under grant number NSF GJ 408 and the Atomic Energy Commission under grant number AT (04-3) 326, PA 30.

ROUNDOFF ERROR ANALYSIS OF THE FAST FOURIER TRANSFORM

Abstract. This paper presents an analysis of roundoff errors occurring in the floating-point computation of the fast Fourier transform. Upper bounds are derived for the ratios of the root-mean-square (RMS) and maximum roundoff errors in the output data to the RMS value of the input data for both single and multidimensional transformations. These bounds are compared experimentally with actual roundoff errors.

1. Introduction. The fast Fourier transform (FFT) is a very efficient algorithm for computing

$$(1.1) \qquad y(j) = \sum_{k=0}^{N-1} e^{i2\pi jk/N} x(k) \qquad (j = 0,1,\ldots,N-1) ,$$

where $\{x(k)\}$ is a given set of complex numbers and $i = \sqrt{-1}$. Let $\underset{\sim}{y}' = (y(0),\ldots,y(N-1))$ and $fl(y)$ be the floating-point representation of Y . In this paper we derive bounds for

$$\|fl(y) - \underset{\sim}{y}\|_{RMS} / \|x\|_{RMS} \quad \text{and} \quad \|fl(\underset{\sim}{y}) - \underset{\sim}{y}\|_{\infty} / \|\underset{\sim}{x}\|_{RMS} ,$$

where

$$\|\underset{\sim}{z}\|_{RMS} = \left(\left(\sum_{k} |z(k)|^2\right)/N\right)^{1/2} \quad \text{and} \quad \|\underset{\sim}{z}\|_{\infty} = \max_{k} |z(k)| .$$

These bounds include the effect of roundoff in computing sines and cosines and are obtained for both single and multidimensional transformations. Special consideration is given to cases when N is a multiple of 2 or 4 .

The subject of roundoff error in the FFT has been studied and reported by others but with less generality or using a different approach. By comparing upper bounds, Gentleman and Sande [1] show that accumulated floating-point roundoff error is significantly less when one uses the FFT than when one computes (1.1) directly. In [2] Welch derives approximate upper and lower bounds on the RMS error in a fixed-point power-of-two algorithm. Weinstein [3] uses a statistical model for floating-point roundoff errors to predict the output-noise variance. Liu and Kaneko [4] also use a statistical approach to predict the roundoff error in a floating-point transformation.

In the following sections, (1) the FFT algorithm is analyzed from the point of view of matrix factorization, (2) error bounds are derived, and (3) experimental comparisons of actual errors with error bounds are presented.

2. The Fast Fourier Transform. In 1965 Cooley and Tukey [5] introduced the algorithm now known as the fast Fourier transform. In this algorithm for computing (1.1) the number of operations required is proportionalto $N \log N$ rather than N^2 . A close look at (1.1) shows that it is precisely the matrix-vector equation $y = Tx$ with the Nth- order matrix T defined by $T(j,k) = e^{i2\pi jk/N}$ $(j,k = 0,1,\ldots,N-1)$. Others have pointed out this fact and have observed that the speedup of the fast Fouriertransform is due to the factorization of T into a small number of sparse matrices [6], [7], [8], [9]. The factorization of T is derived below and is shown to be that given by the following theorem:

THEOREM 1. If T is a matrix of order N with complex exponential elements $T(j,k) = \exp(i2\pi jk/N)$ $(j,k = 0,1,\ldots,N-1)$ and if $N = N_1 N_2 \ldots N_M$, then

$$T = P_{M+1}(T_M P_M)(D_{M-1}\, T_{M-1}\, P_{M-1}) \cdots (D_1 T_1 P_1),$$

where P_ℓ $(I = 1,2,\ldots,M+1)$ are permutation matrices, D_ℓ $(\ell = 1,2,\ldots,M-1)$ are diagonal matrices of complex exponential elements, and T_ℓ $(a = 1,2,\ldots,M)$ are block-diagonal matrices whose blocks have elements $\exp(i2\pi j_\ell k_\ell/N_\ell)$ $(j_\ell, k_\ell = 0,1,\ldots,N_\ell-1)$.

Proof. Following Gentleman and Sande [1] we use the notation $\underline{e}(\theta)$ for $\exp(i2\pi\theta)$. Note that $\underline{e}(\theta_1 + \theta_2) = \underline{e}(\theta_1)\underline{e}(\theta_2)$ and $e(Q) = 1$ if θ is an integer.

Let the indices in (1.1) be expressed as $j = j_1 + j_1^* N_1$ and $k = k_1^* + k_1 N_1^*$ $(j_1,k_1 = 0,1,\ldots,N_1-1 \; ; \; j_1^*,k_1^* = 0,1,\ldots,N_1^*-1)$, where $N_1^* = N_2 N_3 \ldots N_M$. Then one can write

$$(2.1) \qquad y(j_1 + j_1^* N_1) = \sum_{k_1^*} \underline{e}(j_1^* k_1^*/N_1^*) z_1(j_1 + k_1^* N_1) \quad ,$$

where

$$z_1(j_1 + k_1^* N_1) = \underline{e}(j_1 k_1^*/N) \sum_{k_1} \underline{e}(j_1 k_1/N_1) x(k_1^* + k_1 N_1^*) \quad .$$

Let $x^P(k_1 + k_1^* N_1) = x(k_1^* + k_1 N_1^*)$. Then $\underset{\sim}{z}_1 = D_1 T_1 P_1 \underset{\sim}{x}$, where D_1 is a diagonal matrix of complex exponentials, T_1 is the block-diagonal matrix with block elements $\underline{e}(j_1 k_1/N_1)$ $(j_1,k_1 = 0,1,\ldots,N_1-1)$, and P_1 is the permutation matrix defined by $\underset{\sim}{x}^P = P_1 \underset{\sim}{x}$.

Next let indices in (2.1) be expressed as $j_1^* = j_2 + j_2^* N_2$ and $k_1^* = k_2^* + k_2 N_2^*$ $(j_2,k_2 = 0,1,\ldots,N_2-1 \; ; \; j_2^*,k_2^* = 0,1,\ldots,N_2^*-1)$, where $N_2^* = N_3 N_4 \ldots N_M$. Then (2.1) becomes

$$y(j_1 + j_2 N_1 + j_2^* N_1 N_2) = \sum_{k_2^*} \underline{e}(j_2^* k_2^*/N_2^*) z_2(j_2 + j_1 N_2 + k_2^* N_1 N_2) \; ,$$

where

$$z_2(j_2 + j_1 N_2 + k_2^* N_1 N_2) =$$

$$\underline{e}(k_2^*(j_1 + j_2 N_1)/N) \sum_{k_2} \underline{e}(j_2 k_2/N_2) z_1(j_1 + k_2^* N_1 + k_2 N/N_2).$$

Let $z_1^D(k_2 + j_1N_2 + k_2^* N_1 N_2) = z_1(j_1 + k_2^* N_1 + k_2 N/N_2)$. Then $\underset{\sim}{z}_2 = D_2 T_2 P_2 \underset{\sim}{z}_1$, where D_2 and T_2 satisfy the conditions of the theorem and P_2 is defined by $\underset{\sim}{z}_1^P = P_2 \underset{\sim}{z}_1$.

Continuing in this manner one finally arrives at

$$y(j_1 + j_2N_1 + \cdots j_M N_1 N_2 \cdots N_{M-1})$$

$$= \sum_{k_M} e(j_M k_M/N_M) z_{M-1}(j_{M-1} + j_{M-2}N_{M-1} + \cdots j_1 N_2 N_3 \cdots N_{M-1} + k_M N_1 N_2 \cdots) ,$$

where'

$$\underset{\sim}{z}_{M-1} = D_{M-1} T_{M-1} P_{M-1} E_{M-2} \quad .$$

We define P_M and P_{M+1} by $\underset{\sim}{z}_{M-1}^P = P_M \underset{\sim}{z}_{M-1}$ and $\underset{\sim}{y} = P_{M+1} \underset{\sim}{y}^P$, where

$$z_{M-1}^P(k_M + j_{M-1}N_M + \cdots$$

$$\cdot z_{M-1}(j_{M-1} \cdot j_{M-2}N_{M-1} \cdots j_1 N_2 N_3 \cdots N_{M-1} + k_M N_1 N_2 \cdots N_{M-1})$$

and

$$y(j_1 + j_2N_1 + \cdots j_M N_1 N_2 \cdots) \; P_y \; (j_M + j_{M-1}N_M + \cdots N_M) .$$

Then

$$y = P_{M+1} T_M P_M \underset{\sim}{z}_{M-1} ,$$

$$= P_{M+1}(T_M P_M)(D_{M-1} T_{M-1} P_{M-1})\underset{\sim}{z}_{M-2} ,$$

$$= \ldots ,$$

$$= P_{M+1}(T_M P_M)(D_{M-1} T_{M-1} P_{M-1}) \ldots (D_1 T_1 P_1)\underset{\sim}{x} ,$$

$$= Tx$$

and the proof is complete.

At this point it is easy to count the number of operations required by the fast Fourier transform. Whereas direct computation of $\underset{\sim}{y} = Tx$ requires N^2 complex multiplications and $N(N-1)$ complex additions, it is seen that computation of $\underset{\sim}{y} = P_{M+1}(T_M P_M)(D_{M-1} T_{M-1} P_{M-1}) \ldots (D_1 T_1 P_1)\underset{\sim}{x}$ requires $N(M-1 + \sum_{\ell=1}^{M} N_\ell)$ complex multiplications and $N(\sum_{\ell=1}^{M} (N_\ell - 1))$ complex additions.

One further observation should be made before proceeding to the error analysis. This regards a variation of the fast Fourier transform known as the Sande-Tukey algorithm in difference to the Cooley-Tukey algorithm derived above (see [1]). In a matrix factorization corresponding to the Sande-Tukey algorithm, the theorem still holds but with different diagonal matrices D_ℓ $(\ell = 1,2,\ldots,M-1)$. Table 1 compares elements of the diagonal matrices for the two versions.

Table 1. FFT Diagonal Matrix Elements

Cooley-Tukey

D_1 : $e(k_2 j_1 / N_1 N_2)$

⋮

D_{M-2} : $e(k_{M-1}(j_1 + j_2 N_1 + \cdots + j_{M-2} N_1 N_2 \cdots N_{M-3})/N_1 N_2 \cdots$ ●

D_{M-1} : $e(k_M(j_1 + j_2 N_1 + \cdots + j_{M-1} N_1 N_2 \cdots N_{M-2})/N)$

Sande-Tukey

D_1 : $e(j_1(k_M + k_{M-1} N_M + \cdots + k_2 N_3 N_4 \cdots$ ●

⋮

D_{M-2} : $e(j_{M-2}(k_M + k_{M-1} N_M)/N_{M-2} N_{M-1} N_M)$

D_{M-1} : $e(j_{M-1} k_M / N_{M-1} N_M)$

3. Roundoff Errors in the Fast Fourier Transform. In this section we first explain the roundoff error models used and then state and prove a theorem bounding the RMS and maximum errors.

It is assumed that the floating-point accumulator of the computer on which the fast Fourier transform is implemented has at least one digit of extra length (a guard digit). Then the floating-point sum and floating-point product of two floating-point numbers a and b are given by

$$(3.1) \qquad fl(a + b) = (a + b)(1 + \theta\varepsilon)$$

and

$$(3.2) \qquad fl(ab) = ab(1 + \theta\varepsilon) ,$$

where ε is a computer-dependent constant and θ is a generic variable usually different in value at each occurrence but always within the range -1 to 1. (The relative error constant, ε, is $0.5\beta^{1-t}$ for rounded operations or β^{1-t} for chopped operations on a computer, where β is the floating-point computing system base and t is the number of base-p digits in the mantissa of the floating-point number. For example, $\varepsilon = 16^{-5}$ in short-precision floating-point operations on the IBM/360.)

To represent roundoff in computing sines and cosines we introduce an absolute error constant $\gamma > 0$ such that

$$fl(\sin(fl(a))) = \sin(a) + \gamma\theta\varepsilon$$

and

$$fl(\cos(fl(a))) = \cos(a) + \gamma\,\theta\,\varepsilon \ ,$$

where θ and ε are above. This constant depends on how sines and cosines and their arguments are computed for a transformation of a given order, but it is independent of the input data.

Let $x' = (x(0),\ldots,x(N-1))$, $y' = (y(0),\ldots,y(N-1))$ and $fl(y)$ be the floating-point representation of y and let $\| \underset{\sim}{z} \|_{RMS} = ((\sum_k |z(k)|^2)/N)^{1/2}$ and $\| \underset{\sim}{z} \|_\infty = \max_k |z(k)|$. Then we have the following:

THEOREM 2. If $\underset{\sim}{y} = T\underset{\sim}{x}$ is computed by a floating-point fast Fourier transform of order $N = N_1 N_2 \ldots N_M$, then

a. $$\| fl(\underset{\sim}{y}) - \underset{\sim}{y} \|_{RMS} / \| \underset{\sim}{x} \|_{RMS} < \sqrt{N}\, K(N,\gamma)\varepsilon + O(\varepsilon^2)$$

and

b. $$\| fl(\underset{\sim}{y}) - \underset{\sim}{y} \|_\infty / \| \underset{\sim}{x} \|_{RMS} < N\ K(N,\gamma)\varepsilon + O(\varepsilon^2) \ ,$$

where

$$K(N,\gamma) = \sum_{\ell=1}^{M} \alpha(N_\ell) + (M-1)(3 + 2\gamma)$$

and

$$\alpha(N_\ell) = \begin{cases} \sqrt{2} & (N_\ell = 2) \\ 5 & (N_\ell = 4) \\ 2\sqrt{N_\ell}\,(N_\ell + \gamma) & \text{otherwise} \end{cases} \ .$$

Proof of a. First consider computation of the inner product $v = \sum_{\ell=1}^{n} a(\ell)u(\ell)$ by the algorithm: begin v := a(1) ⊗ u(1); for ℓ := 2 step 1 until n do v := v + a(ℓ) ⊗ u(ℓ) end where it is known that u is exactly representable in floating-point while $\underset{\sim}{a}$ satisfies $fl(a(\ell)) = a(1) + y\,\theta\,\varepsilon$ ($\ell = 1,2,\ldots,n$) for y , θ and ε as above. By repeated application of (3.1) and (3.2), as in Wilkinson [10], one finds that

$$fl(v) = (a(1) + \gamma\theta\varepsilon)u(1)(1+\theta\varepsilon)^{n} + (a(2)+\gamma\,\theta\,\varepsilon)u(2)(1+\theta\varepsilon)^{n} +$$

$$+ (a(3)+\gamma\,\theta\,\varepsilon)u(3)(1+\theta\varepsilon)^{n-1} + \ldots + a(n)(1+\gamma\,\theta\,\varepsilon)u(n)(1+\theta\varepsilon)^{2} .$$

Expanding factors $(1+\theta\varepsilon)^{\ell}$ and regrouping terms, this becomes

$$fl(v) = v + \varepsilon[(a(1)n\theta+\gamma\theta)u(1) + (a(2)n\theta+\gamma\theta)u(2) +$$

$$+ (a(3)(n-1)\theta+\gamma\theta)u(3) + \ldots + (a(n)2\theta+\gamma\theta)u(n)] + O(\varepsilon^{2}) ,$$

where $O(\varepsilon^{2})$ includes all terms of order ε^{2} . Thus, it follows that floating-point computation of the matrix-vector product v = Au , where $fl(A(j,\ell)) = A(j,\ell) + \gamma\,\theta\,\varepsilon$ and $fl(u(\ell)) = u(\ell)$, is given exactly by

(3.3)

$$\begin{bmatrix} fl(v(1)) \\ fl(v(2)) \\ \vdots \\ fl(v(n)) \end{bmatrix} = \begin{bmatrix} v(1) \\ v(2) \\ \\ v(n) \end{bmatrix} + \varepsilon \begin{bmatrix} A(1,1)n\theta+\gamma\theta & A(1,2)n\theta+\gamma\theta & \cdots & A(1,n)2\theta+\gamma\theta \\ A(2,1)n\theta+\gamma\theta & A(2,2)n\theta+\gamma\theta & \cdots & A(2,n)2\theta+\gamma\theta \\ & & \vdots & \\ A(n,1)n\theta+\gamma\theta & A(n,2)n\theta+\gamma\theta & \cdots & A(n,n)2\theta+\gamma\theta \end{bmatrix} \begin{bmatrix} u(1) \\ u(2) \\ \vdots \\ u(n) \end{bmatrix} + \begin{bmatrix} O(\varepsilon^2) \\ O(\varepsilon^2) \\ \vdots \\ O(\varepsilon^2) \end{bmatrix}$$

Next consider computation of (1.1) without using the FFT. We write this complex computation as its real equivalent:

$$\begin{bmatrix} \underset{\sim}{y}_R \\ \hline \underset{\sim}{y}_I \end{bmatrix} = \left[\begin{array}{c|c} C & -S \\ \hline S & C \end{array}\right] \begin{bmatrix} \underset{\sim}{x}_R \\ \hline \underset{\sim}{x}_I \end{bmatrix}$$

where C and S are real matrices with elements $C(j,k) = \cos(2\pi(j-1)(k-1)/N)$ and $S(j,k) = \sin(2\pi(j-1)(k-1)/N)$ $(j,k = 1,2,\ldots,N)$, and $\underset{\sim}{x}_R$, $\underset{\sim}{x}_I$, $\underset{\sim}{y}_R$, $\underset{\sim}{y}_I$ are the real and imaginary parts of x and y . Note that the RMS value of a complex vector is $\sqrt{2}$ times as large as the RMS value of its real equivalent and that the RMS value of any vector is a multiple of the Euclidean norm and therefore is consistent with the same matrix norms as the Euclidean norm. [I.e., If $v = A\underset{\sim}{u}$, then $v_{RMS} \leq \| A \| u_{RMS}$, where $\| A \|$ is the Frobenius norm (the square root of the sum of the squared-magnitudes of all elements) or the spectral norm (the square root of the largest eigenvalue of A^*A). See Wilkinson [10] or Isaacson and Keller [11].] Therefore, by (3.3) and the properties of norms,

$$(3.4) \qquad \| fl(\underset{\sim}{y}) - \underset{\sim}{y} \|_{RMS} \leq \varepsilon \| M \| \, \| \underset{\sim}{x} \|_{RMS} + O(\varepsilon^2) \ ,$$

where M is the matrix of Figure 1. Using the fact that $|C(j,k)|^2 + |S(j,k)|^2 = 1$, the Frobenius norm of M is bounded by

$$(3.5) \qquad \| M \| \leq \{N[(2N)^2 + (2N)^2 + (2N-1)^2 + \ldots + 3^2 + 2^2]\}^{1/2} + 2N\gamma$$

$$< 2N(N+\gamma)$$

when N is greater than 2 .

Finally we analyze the fast Fourier transform. Let $\underset{\sim}{z}_1 = D_1 T_1 P_1 \underset{\sim}{x}$. Since the permutation matrix simply reorders vector values, it introduces no roundoff error. Assume $fl(\mathbf{x}) = \underset{\sim}{x}$. Then

$$(3.6) \qquad fl(\underset{\sim}{z}_1) - \underset{\sim}{z}_1 = fl(D_1 \, fl(T_1 P_1 \underset{\sim}{x})) - D_1 T_1 P_1 \underset{\sim}{x}$$

$$= fl(D_1 \, fl(v)) - D_1 \, fl(\underset{\sim}{v}) + D_1[fl(T_1 \underset{\sim}{u}) - T_1 \underset{\sim}{u}],$$

where $u = P_1 \underset{\sim}{x}$ and $\underset{\sim}{v} = T_1 \underset{\sim}{u}$. To bound $fl(T_1 \underset{\sim}{u}) - T_1 \underset{\sim}{u}$, recall that T_1 is a block-diagonal matrix whose blocks are Fourier transform matrices of order N_1 . Let $\underset{\sim}{u}_\ell$, $\underset{\sim}{v}_\ell$ $(\ell = 1,2,\ldots,N/N_1)$ be N_1-vectors such that

$$\underset{\sim}{u} = \begin{bmatrix} \underset{\sim}{u}_1 \\ \underset{\sim}{u}_2 \\ \vdots \\ \underset{\sim}{u}_{N/N_1} \end{bmatrix} \quad \text{and} \quad \underset{\sim}{v} = \begin{bmatrix} \underset{\sim}{v}_1 \\ \underset{\sim}{v}_2 \\ \vdots \\ \underset{\sim}{v}_{N/N_1} \end{bmatrix} \ .$$

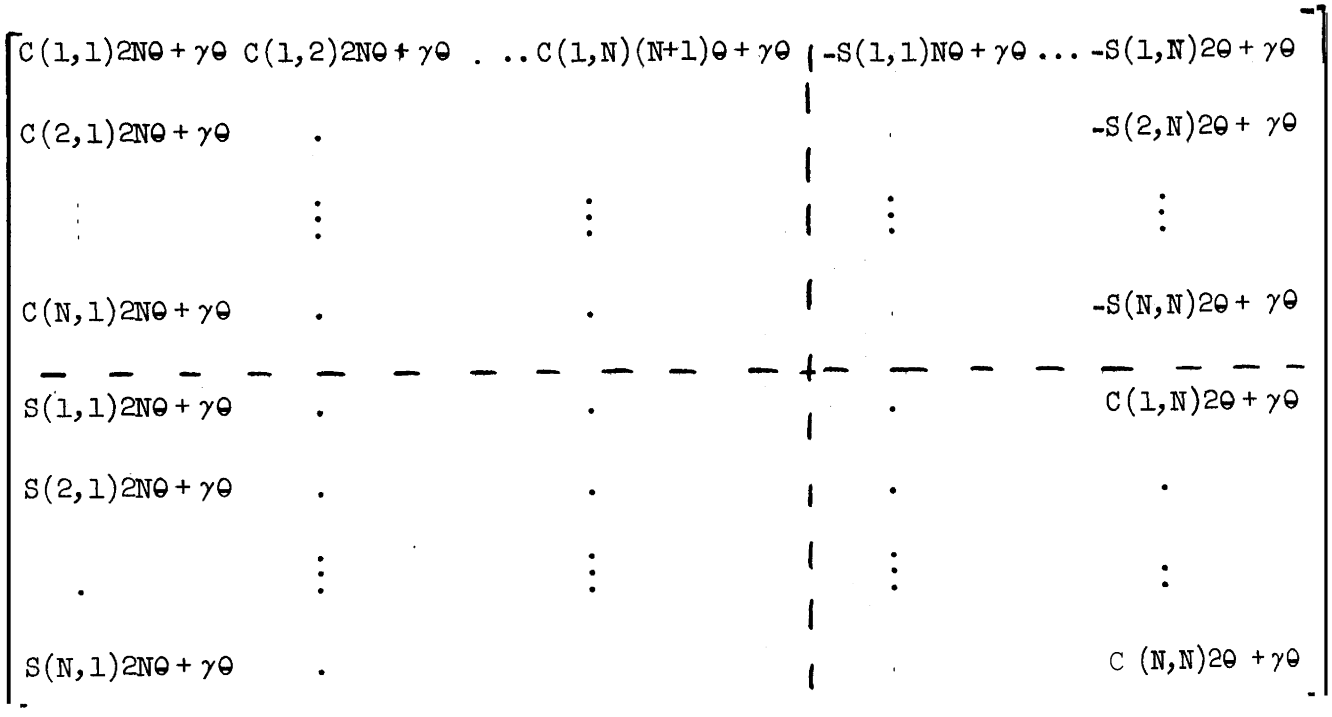

Figure 1. Direct Transformation Error Matrix

Then by (3.4) and (3.5), $\| \text{fl}(\underset{\sim}{v}_\ell) - \underset{\sim}{v}_\ell \|_{RMS} \leq \varepsilon \quad 2N_1(N_1+\gamma)\|\underset{\sim}{u}_\ell\|_{RMS} + O(\varepsilon^2)$ ($\ell = 1,2,\ldots,N/N_1$) when N_1 is greater than 2 . If $N_1 = 2$, this inequality still holds. In fact, we can do much better. Figures 2 and 3 show the block-diagonal factor matrices for the cases when N has factors 2 or 4. By inspection, one can see that in these cases no sines and cosines are computed, no multiplications are required, and there are only N_ℓ elements to be summed as compared with $2N_\ell$-1 in other cases. Thus,one can easily show that

$$\|\text{fl}(\underset{\sim}{v}_\ell) - \underset{\sim}{v}_\ell\|_{RMS} \leq \varepsilon \sqrt{N_1}\, \alpha(N_1)\|\underset{\sim}{u}_\ell\|_{RMS} + O(\varepsilon^2) \qquad (\ell = 1,2,\ldots,N/N_1) ,$$

where

$$\alpha(N_1) = \begin{cases} \sqrt{2} & (N_1 = 2) \\ 5 & (N_1 = 4) \\ 2\sqrt{N_1}\,(N_1+\gamma) & \text{otherwise} \end{cases}$$

It immediately follows that

$$(3.7) \qquad \|\text{fl}(T_1 \underset{\sim}{u}) - T_1 \underset{\sim}{u}\|_{RMS} \leq \varepsilon \sqrt{N_1}\, \alpha(N_1)\|\underset{\sim}{u}\|_{RMS} + O(\varepsilon^2)$$

for $\alpha(N_1)$ as above.

In the same way we obtain a bound on the error in multiplication by the complex-diagonal matrix D_1 . The bound is given by

$$(3.8) \qquad \|\text{fl}(D_1\, \text{fl}(\underset{\sim}{v})) - D_1\, \text{fl}(\underset{\sim}{v})\|_{RMS} \leq \varepsilon(2\sqrt{2}+2\gamma)\,\|\text{fl}(\underset{\sim}{v})\|_{RMS} + O(\varepsilon^2) .$$

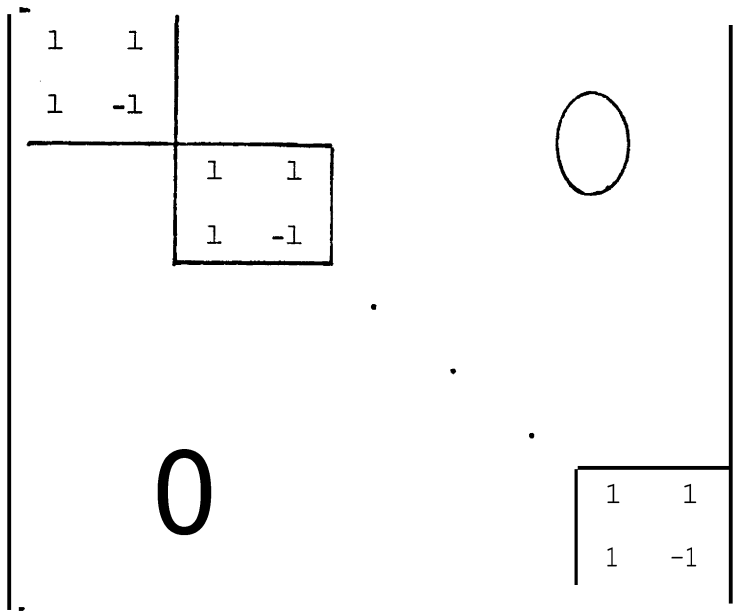

Figure 2. The Block-Diagonal Factor Matrix with 2nd-Order Blocks.

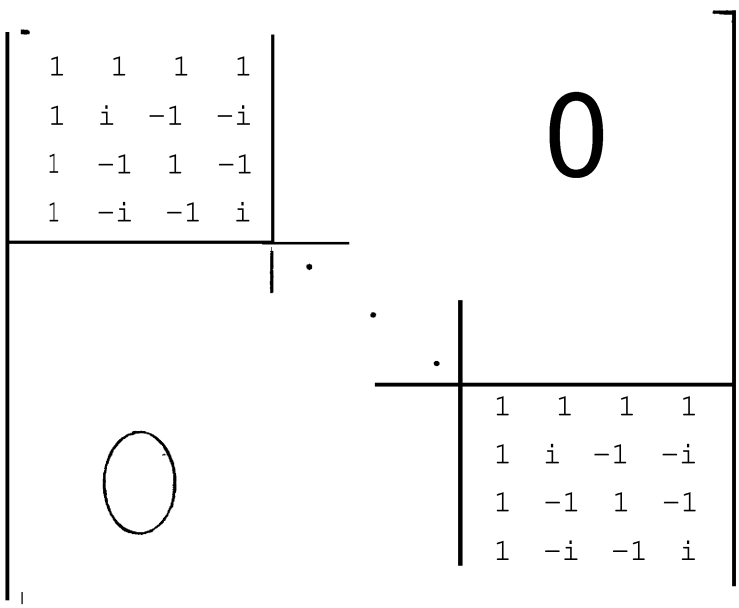

Figure 3. The Block-Diagonal Factor Matrix with 4-th Order Blocks.

From (3.7) it follows that $\|\mathrm{fl}(\underset{\sim}{y})\|_{RMS} = \|\underset{\sim}{y}\|_{RMS} + O(\varepsilon)$. Furthermore, the spectral norms of D_1 , T_1, and P_1 are 1 , $\sqrt{N_1}$, and 1, respectively, since $D_1^* D_1 = I$, $T_1^* T_1 = N_1 I$ and $P_1^* P_1 = I$, where I is the N by N identity matrix. So from (3.6),(3.7) and (3.8) we get

$$\|\mathrm{fl}(\underset{\sim}{z}_1) - \underset{\sim}{z}_1\|_{RMS} \leq \varepsilon \sqrt{N_1}\,(\alpha(N_1) + 3 + 2\gamma)\|\underset{\sim}{x}\|_{RMS} + O(\varepsilon^2) \quad ,$$

where $\alpha(N_1)$ is given above.

The next step is to let $\underset{\sim}{z}_2 = D_2 T_2 P_2 \underset{\sim}{z}_1$. Then

$$\mathrm{fl}(\underset{\sim}{z}_2) - \underset{\sim}{z}_2 = \mathrm{fl}(D_2 T_2 P_2\, \mathrm{fl}(\underset{\sim}{z}_1)) - D_2 T_2 P_2\, \mathrm{fl}(\underset{\sim}{z}_1) + D_2 T_2 P_2 [\mathrm{fl}(\underset{\sim}{z}_1) - \underset{\sim}{z}_1]$$

and

$$\|\mathrm{fl}(\underset{\sim}{z}_2) - \underset{\sim}{z}_2\|_{RMS} \leq \varepsilon (N_1 N_2)^{1/2} (\alpha(N_1) + \alpha(N_2) + 2(3 + 2\gamma))\|\underset{\sim}{x}\|_{RMS} + O(\varepsilon^2) .$$

The proof of part a. is completed by continuing in this manner and using Theorem 1.

Proof of b. The proof is extremely simple. Let $e(j) = \mathrm{fl}(y(j)) - y(j)$. Then

$$\max_j |e(j)|^2 \leq \sum_{j=0}^{N-1} |e(j)|^2 \quad ,$$

from which it follows that

$$\max_j |e(j)| \leq \sqrt{N}\, \|\underset{\sim}{e}\|_{RMS} \quad .$$

Substituting the bound of part a for $\|\underset{\sim}{e}\|_{RMS}$ completes the proof.

It is not necessary to obtain a bound on the maximum error by using part a. Instead one can use matrix infinity norms in the same fashion that matrix spectral norms were used above. But the infinity norms of the factor matrices, T_ℓ, are proportional-to N_ℓ rather than $\sqrt{N_\ell}$, and so a higher bound results.

4. Roundoff Errors in Multidimensional Transformations. The efficiency of the fast Fourier transform has made it economically feasible to compute higher dimensional Fourier transformations in applications such as picture processing and x-ray diffraction studies. In this section, bounds on roundoff errors in multidimensional FFTs are derived.

The problem is to bound roundoff errors in computing

$$(4.1)\quad Y(t_1,t_2,\ldots,t_m) =$$

$$= \sum_{s_1} \sum_{s_2} \cdots \sum_{s_m} \underline{e}(s_1t_1/N_1 + s_2t_2/N_2 + \ldots + s_mt_m/N_m)X(s_1,s_2,\ldots,s_m)$$

$$(s_\ell,t_\ell = 0,1,\ldots,N_\ell-1;\ \ell = 1,2,\ldots,m) .$$

Let

$$E(t_1,t_2,\ldots,t_m) = fl(Y(t_1,t_2,\ldots,t_m)) - Y(t_1,t_2,\ldots,t_m) ,$$

$$[fl(Y) - Y]_{RMS} = \left\{\left(\sum_{t_1} \sum_{t_2} \cdots \sum_{t_m} |E(t_1,t_2,\ldots,t_m)|^2\right) \Big/ N_1N_2\ldots N_m\right\}^{1/2},$$

and

$$[fl(Y) - Y]_{MAX} = \max_{t_1,t_2,\ldots,t_m} |E(t_1,t_2,\ldots,t_m)| .$$

Then we have:

THEOREM 3. The RMS and maximum error due to roundoff in a multidimensional fast Fourier transform are bounded by

$$\text{a.} \quad [\mathrm{fl}(Y) - Y]_{RMS}/X_{RMS} \leq \varepsilon(N_1 N_2 \ldots N_m)^{1/2} \sum_{\ell=1}^{m} K(N_\ell, \gamma) + O(\varepsilon^2)$$

and

$$\text{b.} \quad [\mathrm{fl}(Y) - Y]_{MAX}/X_{RMS} \leq \varepsilon\, N_1 N_2 \ldots N_m \sum_{\ell=1}^{m} K(N_\ell, \gamma) + O(\varepsilon^2),$$

where $K(N_\ell, \gamma)$ (ℓ = 1,2, [illegible]) is the error constant given in Theorem 2.

Proof. Let (4.1) be rewritten as the system of equations

$$Z_{\ell-1}(s_1, \ldots, s_{\ell-1}, t_\ell, \ldots, t_m) = \sum_{s_\ell} e(s_\ell t_\ell / N_\ell) Z_\ell(s_1, \ldots, s_\ell, t_{\ell+1}, \ldots, t_m)$$

$$(\ell = 1, 2, \ldots, m)$$

with $Z_0 = Y$ and $Z_m = X$, and describe this system of equations by the notation

$$Z_{\ell-1} = T_\ell Z_\ell \qquad (\ell = 1, 2, \ldots, m) .$$

Then by adding and subtracting identical terms to the equation

$$\mathrm{fl}(Y) - Y = \mathrm{fl}(T_1\, \mathrm{fl}(T_2 \ldots \mathrm{fl}(T_m X) \ldots)) - T_1 T_2 \ldots T_m X$$

one gets

$$\begin{aligned} fl(Y) - Y &= fl(T_1\, fl(Z_1)) - T_1\, fl(Z_1) \\ &+ T_1\, fl(T_2\, fl(Z_2)) - T_1T_2\, fl(Z_2) \\ &+ T_1T_2\, fl(T_3\, fl(Z_3)) - T_1T_2T_3\, fl(Z_3) \\ &+ \cdots \\ &+ T_1T_2\cdots T_{m-1}\, fl(T_mX) - T_1T_2\cdots T_mX \quad . \end{aligned}$$

Now take the RMS value of both sides and use the Cauchy-Schwartz inequality to get

$$\begin{aligned} (4.2)\quad [fl(Y) - Y]_{RMS} &\le [fl(T_1\, fl(Z_1)) - T_1\, fl(Z_1)]_{RMS} \\ &+ [T_1[\, fl(T_2\, fl(Z_2)) - T_2\, fl(Z_2)\,]]_{RMS} \\ &+ \cdots + \\ &+ [T_1T_2\cdots T_{m-1}[fl(T_mX) - T_mX]]_{RMS} \quad . \end{aligned}$$

Using Theorem 2 it is not difficult to prove that

$$(4.3)\qquad [fl(Z_{\ell-1}) - Z_{\ell-1}]_{RMS}/[Z_\ell]_{RMS} \le \varepsilon\sqrt{N_\ell}\;\; K(N_\ell,\gamma) + O(\varepsilon^2)$$

Nor is it difficult to prove

$$(4.4)\qquad [Z_{\ell-1}]_{RMS} = \sqrt{N_\ell}\;[Z_\ell]_{RMS} \quad .$$

Therefore, by (4.2),(4.3) and (4.4)

$$[\mathrm{fl}(Y) - Y]_{RMS} \leq \varepsilon\{(N_1)^{1/2} K(N_1,\gamma)[\mathrm{fl}(Z_1)]_{RMS} + (N_1N_2)^{1/2} K(N_2,\gamma)[\mathrm{fl}(Z_2)]_{RMS} + \ldots + (N_1N_2\ldots N_m)^{1/2} K(N_m,\gamma)[\mathrm{fl}(X)]_{RMS}\} + O(\varepsilon^2) \quad .$$

But by (4.3) $[\mathrm{fl}(Z_\ell)]_{RMS} = [Z_\ell]_{RMS} + O(\varepsilon)$ $(\ell = 1,2,\ldots,m-1)$, and by (4.4) $[Z_\ell]_{RMS} \leq (N_{\ell+1}N_{\ell+2}\ldots N_m)^{1/2} [X]_{RMS}$. Assuming that $[\mathrm{fl}(X)]_{RMS} = [X]_{RMS}$, or at least $[\mathrm{fl}(X)]_{RMS} = [X]_{RMS} + O(\varepsilon)$, the proof of part a. is complete.

Part b is proved by arguments identical to those used in the proof of part b of Theorem 2.

5. Experimental Results. Roundoff error bounds are always pessimistic -- sometimes so much so that they give no indication of the true error, To find out how pessimistic the error bounds of Section 3 are, the following experiment was performed. Using two different FORTRAN programs, one by N. M. Brenner [12] and the other by R. C. Singleton [13], a mixed radix fast Fourier transform of Gaussian data with mean 0 and variance 2 was computed in both short and long precision on the Stanford IBM 360/67. The actual error was computed as the difference between the short precision results and the truncated long precision results. The constant γ used in determining the error bound was computed by taking the difference between short precision and truncated long precision numbers representing sines and cosines. The results of this experiment are given in Table 2. Note that the RMS error bound is roughly 20 times larger than the RMS error and the MAX error bound is roughly 2 orders of magnitude larger than the MAX error. Also note the relative size of the error bounds with respect to values of the transformed data. Even though the bounds are pessimistic they might be used as a threshold for deciding what confidence to place in transformed data of relatively small magnitude.

Table 2

Comparison of Actual Errors with Error Bounds

Order of Transform. and Factorization	Values of Transformed Data				Errors in Transformed Data		A Priori Bounds on Errors	
	MIN	RMS	MAX	Y	RMS	MAX	RMS	MAX
128 = 4 4 4 2 *	0.9543	16.54	36.13	3.1	0.000032	0.000082	0.000698	0.007897
128 = 4 2 2 2 4**	0.9543	16.54	36.13	1.7	0.000026	0.000064	[illegible]	0.007138
256 = 4 4 4 4 *	1.4436	21.78	53.48	3.1	0.000047	0.000153	0.000992	0.015875
256 = 4 4 4 4 **	1.4436	21.78	53.48	4.7	0.000070	0.000216	0.001187	0.018992
512 = 4 4 4 4 2*	1.4158	31.20	81.04	4.2	0.000101	0.000306	0.001994	0.045121
512 = 4 4 2 4 4**	1.4158	31.20	81.04	4.6	0.000106	0.000307	0.002083	0.047141
1024 = 4 4 4 4 4*	2.2109	44.38	130.41	9.3	0.000202	0.000648	0.004720	0.151041
1024 = 4 4 4 4 4**	2.2110	44.38	130.41	8.9	0.000291	0.001163	0.004572	0.146301
100 = 4 5 5 *	1.5535	14.98	29.17	5.2	0.000129	0.000491	0.001755	0.017554
100 = 5 4 5 **	1.5534	14.98	29.17	7.7	0.000043	0.000122	0.002218	0.022176
200 = 4 2 5 5 *	1.3670	19.50	45.60	6.8	0.000175	0.000560	0.003014	0.042628
200 = 5 2 2 2 5**	1.3670	19.50	45.60	3.4	0.000046	0.000109	0.002223	0.031432
300 = 4 3 5 5 *	0.6539	23.64	54.42	8.1	0.000239	0.000663	0.004905	0.084952
300 = 5 2 3 2 5**	0.6539	23.64	54.42	7.0	0.000098	0.000301	0.004802	0.083172
400 = 4 4 5 5 *	2.8367	27.50	66.63	7.1	0.000243	0.000743	0.004440	0.088793
400 = 4 5 5 4 **	2.8368	27.50	66.63	7.7	0.000120	0.000430	0.004685	0.093692

Table 2 (continued)

Order of Transform and Factorization		Values of Transformed Data				Errors in Transformed Data		A Priori Bounds on Errors	
		MIN	RMS	MAX	γ	RMS	MAX	RMS	MAX
$125 = 5\ 5\ 5$	*	0.6356	16.42	37.31	4.2	0.000161	0.000558	0.002304	0.025765
$125 = 5\ 5\ 5$	**	0.6355	16.42	37.31	3.1	0.000037	0.000085	0.001998	0.022333
$243 = 3\ 3\ 3\ 3\ 3$	*	0.0957	21.24	53.30	4.1	0.000171	0.000424	0.003379	0.052672
$243 = 3\ 3\ 3\ 3\ 3$	**	0.0957	21.24	53.30	9.0	0.000089	0.000323	0.005911	0.092140
$343 = 7\ 7\ 7$	*	0.9315	25.67	60.21	7.4	0.000160	0.000536	0.006487	0.120138
$343 = 7\ 7\ 7$	**	0.9315	25.67	60.21	7.9	0.000123	0.000384	0.006700	0.124080

* Brenner's Program

** Singleton's Program

6. Conclusion. In the preceding sections roundoff errors in the floating-point fast Fourier transform have been analyzed. Bounds on RMS and maximum errors in transformed data were determined for both single and multidimensional transforms, and in the case of a one-dimensional transform results of a computational experiment show how close these bounds are to the actual roundoff errors. The bounds include the effect of roundoff in computing sines and cosines and, if contributions to the actual errors are in the same proportion as to the error bounds, a close look at the error bounds shows that the effect of roundoff in computing sines and cosines is not negligible but in fact contributes the same order of magnitude to the total error as the roundoff in additions and multiplications.

So far nothing has been said about floating-point representation of input data. It was assumed that these numbers were exactly representable in machine precision. If not, an additional term must be added to the roundoff error to account for rounding input data. Suppose $fl(\underset{\sim}{x}) = \underset{\sim}{x}+\underset{\sim}{\delta}$. Then the additional term is

$$\|T\underset{\sim}{\delta}\|_{RMS} \leq \sqrt{N}\, \|\underset{\sim}{\delta}\|_{RMS} .$$

On the-other hand, suppose that the input data is known to a number of significant digits fewer than that of machine precision. For example, the data might have come from an analog device of limited accuracy. Then the bounds on roundoff error can be used in reverse as suggested by the following: Let the roundoff error be given exactly by the complex N-vector e . This vector can be considered the exact solution of the

equation $e = T\delta$ for some fictional δ bounded by

$$\|\delta'\|_{RMS} = \|e\|_{RMS} / \sqrt{N}$$

$$\leq \varepsilon\, K(N,\gamma) \|x\|_{RMS} + O(\varepsilon^2) \quad ,$$

and

$$\|\delta\|_{\infty} \leq \varepsilon \sqrt{N}\, K(N,\gamma) \|x\|_{RMS} + O(\varepsilon^2) \quad .$$

If it should turn out that $\varepsilon \sqrt{N}\, K(N,\gamma) \|x\|_{RMS}$ is smaller than the least significant digit of the input data, the roundoff error is negligible.

Acknowledgments. The author wishes to thank Professor Gene Golub of Stanford University for his advice and encouragement during the research for this paper. Special thanks also go to Sylvania Electronic Systems, Western Division, for support received while the author was at Stanford. This research was done in partial fulfillment of the requirements for a doctoral degree in Computer Science.

References

[1] W. M. Gentleman and G. Sande, "Fast Fourier transforms -- for fun and profit," 1966 Fall Joint Computer Conference, AFIPS Proc., vol. 29, Washington, D.C.: Spartan, 1966, pp. 563-578.

[2] P. D. Welch, "A fixed-point fast Fourier transform error analysis," IEEE Trans. Audio and Electroacoustics, vol. AU-17, pp. 151-157, June 1969.

[3] C. J. Weinstein, "Roundoff noise in floating point fast Fourier transform computation," IEEE Trans. Audio and Electroacoustics, vol. AU-17, pp. 209-215, September 1969.

[4] B. Liu and T. Kaneko, personal communication.

[5] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Computation, vol. 19, pp. 297-301, April 1965.

[6] W. M. Gentleman, "Matrix multiplication and fast Fourier transforms," Bell Sys. Tech. J., vol. 47, pp. 1099-1103, July-August 1968.

[7] R. C. Singleton, "On computing the fast Fourier transform," Commun. ACM, vol. 10, pp. 647-654, October 1967.

[8] F. Theilheimer, "A matrix version of the fast Fourier transform," IEEE Trans. Audio and Electroacoustics, vol. AU-17, pp. 158-161, June 1969.

[9] D. K. Kahaner, "Matrix Description of the Fast Fourier Transform," Los Alamos Scientific Laboratory Report LA-4275-MS, December 1969.

[10] J. H. Wilkinson, Rounding Errors in Algebraic Processes, Prentice-Hall, Englewood Cliffs, New Jersey, 1963.

[II] E. Isaacson and H. B. Keller, Analysis of Numerical Methods, Wiley, New York, 1966.

[12] N. M. Brenner, "Three FORTRAN programs that perform the Cooley-Tukey Fourier transform," M.I.T. Lincoln Lab., Lexington, Mass., Technical Note 1967-2, July 1967.

[13] R. C. Singleton, "An algorithm for computing the mixed radix fast Fourier transform," IEEE Trans. Audio and Electroacoustics, vol. AU-17, pp. 93-103, June 1969.