Note on a Lower Bound of the Linear Complexity of the Fast Fourier Transform

JACQUES MORGENSTERN

Université de Nice, Nice, France

ABSTRACT. A lower bound for the number of additions necessary to compute a family of linear functions by a linear algorithm is given when an upper bound c can be assigned to the modulus of the complex numbers involved in the computation. In the case of the fast Fourier transform, the lower bound is $(n/2) \log_2 n$ when $c = 1$.

KEY WORDS AND PHRASES: numerical algorithms, fast Fourier transform, matrix product

CR CATEGORIES: 5.12, 5.13

Section 1

We recall the notation of [1] and [2]: A linear algorithm is a sequence $\mathfrak{F}_0, \mathfrak{F}_1, \cdots, \mathfrak{F}_m$ of linear affine functions from $\mathbf{C}^r$ into $\mathbf{C}$, where $\mathfrak{F}_0$ contains the constants and the projections, and such that $\mathfrak{F}_{i+1} = \mathfrak{F}_i \cup \{\lambda_i f + \mu_i g\}$ with $\lambda_i, \mu_i \in \mathbf{C}$ and $f, g \in \mathfrak{F}_i$. Let $\mathfrak{F}$ be a family of n forms in r variables over $\mathbf{C}$, the matrix of its coefficients being F. Let $\alpha = \mathfrak{F}_0, \mathfrak{F}_1, \cdots, \mathfrak{F}_m$ be a linear algorithm computing $\mathfrak{F}(\mathfrak{F} \leq \mathfrak{F}_m)$ with the *minimum* number of additions. Obviously this algorithm also minimizes the number of additions to compute *any* $\mathfrak{F}_i$.

Remark. To my knowledge it is an unsolved problem to know if a nonlinear algorithm would reduce the number of additions to compute a given set of linear functions.

Let $\Delta(F) = \max |d|$ where d is the determinant of any square submatrix of F.

PROPOSITION. *If at each step of the algorithm* $|\lambda_1| \leq c$ *and* $\mu_i \leq c$, $c > \frac{1}{2}$, *then the number* m^+ *of additions is greater than* $(\log |\Delta(F)|)/(\log (2c))$.

PROOF. According to [2] the functions in each $\mathfrak{F}_i$ are homogeneous since α is minimal. The proof then goes by induction on i: The inequality holds for $\mathfrak{F}_0$ (i.e. the projections $\Delta_0 = 1$). Let Δ_i be a subdeterminant of $\mathfrak{F}_i$ of greatest modulus, and let m_i^+, the number of additions necessary to compute $\mathfrak{F}_i$, be such that $m_i^+ \geq (\log |\Delta_i|)/(\log (2c))$ or $(2c)_i^{m^+} \geq \Delta_i$. If Δ_{i+1} is a subdeterminant from $\mathfrak{F}_{i+1}$ of greatest modulus, then two cases can occur: Either Δ_{i+1} does not contain a subrow from $\varphi_i = \lambda_i f + \mu_i g$ and Δ_{i+1} can be taken equal to Δ_i, or Δ_{i+1} contains a subrow of φ_i and then $\Delta_{i+1} = \lambda_i \Delta_i' + \mu_i \Delta_i''$ with $|\lambda_i|$ and $|\mu_i| \leq c$ and $|\Delta_i'|, |\Delta_i''| \leq \Delta_i'$; therefore $|\Delta_{i+1}| \leq 2c \cdot |\Delta_i|$, since $c \geq \frac{1}{2}$, which gives the result for the family $\mathfrak{F}$.

Author's present address: Mathématiques et Sciences Théoriques, Université de Nice, Nice, France.

Section 2

In the case where $\mathfrak{F}$ is the family of the discrete Fourier transforms on $x_1, x_2, \cdots, x_n$, the corresponding matrix F is the Hadamard matrix $f_{k,l}$, with $f_{k,l} = \omega^{(k-1)(l-1)}$, $k, l = 1, 2, \cdots, n$ where $\omega = \exp\ (2i\pi/n)$. The row vectors are pairwise orthogonal in the Hilbert space C^n, with the usual Hermitian form, their common length being $n^{\frac{1}{2}}$; hence taking a basis along those n vectors, we get that $|\Delta| = n^{n/2}$ (see [3]), from which follows $m^+ \geq ((n/2) \log n)/(\log c + \log 2)$.

PROPOSITION. *If one tries to compute the discrete Fourier transform using complex numbers of modulus less than or equal to one, the minimum number of "operations" in the sense of* [4] *is greater than* $(n/2)\ \log_2 n$.

Comment. In another paper, to appear soon, the author will give an example of reduction of the number of operations to compute the discrete Fourier transform of order 7 using complex numbers of larger moduli. It is well known (see [4]) that the discrete Fourier transform can be evaluated by a linear algorithm using constants of modulus 1 within $n \log n$ additions.

A linear algorithm using no other coefficients than the entries x_{ij} and y_{hk} of the $n \times n$ matrices X and Y cannot compute the product $X \cdot Y$ in less than $(n^2/2) \log n$ additions (since the product can be viewed as a product of an $n^2 \times n^2$ matrix and a vector).

REFERENCES

1. MORGENSTERN, J. Algorithmes linéaires. *Compt. Rend. Acad. Sci. 272*, 1059–1060.
2. MORGENSTERN, J. On linear algorithms. In *Theory of Machines and Computations*, Academic Press, New York and London, 1971, pp. 59–66.
3. QUEYSANNE, M. *Agèbre M.G.P. Collection U.* Armand Colin, Paris, 1966, Exercice No. 258, p. 363.
4. COOLEY, J. W , AND TUKEY, J. W. An algorithm for the machine computation of complex Fourier series. *Math. Comp. 19* (1964), 297–301.

RECEIVED JULY 1972; REVISED SEPTEMBER 1972