INFORMATION PROCESSING LETTERS 1 (1971) 66–68. NORTH-HOLLAND PUBLISHING COMPANY

EVALUATING POLYNOMIALS AT MANY POINTS

A. BORODIN and I. MUNRO
Department of Computer Science, University of Toronto, Canada

Received 30 April 1971

polynomial evaluation preconditioning fast matrix multiplication

1. Introduction

Polynomial evaluation is one of the most frequently occurring tasks in numerical computations. In fact, many algorithms entail the evaluation of one or more polynomials at a large number of values. The purpose of this note, is to reflect on this evaluation problem, more from a theoretical than practical point of view. Specifically, we are interested in the number of basic arithmetics (+, −, ×, ÷) required for a set of specified evaluations. For definiteness, we shall restrict our remarks to polynomials in one indeterminant over the field of real numbers, assuming exact arithmetic. Our discussion could easily be extended to multivariate polynomials and (with some care) to evaluation over the integers. The avoidance of roundoff considerations and the consideration of general polynomials of high degree are the reasons why we disclaim practical significance.

2. Some known results

Suppose we wish to evaluate an nth degree polynomial $\Sigma_{i=0}^{n} a_i x^i$ at one point. If nothing is known about the coefficients $\{a_i\}$ in advance, then Ostrowski [5] and Belaga [1] have shown that n additions or subtractions (a/s) are required for the evaluation, and Garcia [3] and Pan [6] have shown that n multiplications or divisions (m/d) are needed. Thus, Horner's rule is optimal in this regard. These results were extended by Winograd [8] who has shown that the evaluation of m polynomials of degree n_i $(1 \leq i \leq m)$ at one point requires $\Sigma_{i=1}^{m} n_i$ a/s and $\Sigma_{i=1}^{m} n_i$ m/d operations.

Another way to formulate the evaluation problem is to count only those operations that involve the indeterminant. That is, we can "precondition" or "adapt" the coefficients $\{a_i\}$ without cost. In fact, we might even allow analytic functions $\mu_i = \phi_i(a_0, \ldots, a_n)$ to occur as parameters in a program for polynomial evaluation. Thus, we are asking how many operations are required when we are allowed to reexpress the polynomial in something other than the standard form. Motzkin [4] introduced this concept of preconditioning, and (incorporating an improvement by Pan [6]) he showed that $\lfloor \frac{1}{2} n \rfloor + 1$ m/d operations were required to evaluate "most" nth degree polynomials, even with preconditioning. Belaga [1] showed that n a/s were still required with preconditioning. The important point is that schemes have been developed which nearly achieve these bounds.

The application should be obvious. If one wishes to evaluate a polynomial at k points, where k is "much greater" than n, the cost of preconditioning can be assumed to be negligible, and thus preconditioning affords a saving of approximately $k \times \frac{1}{2} n$ multiplications. Winograd [8] has also extended these results to the evaluation of several polynomials.

3. A better scheme for non-iterative evaluation

The preceding remarks lead one to believe that if the number of points is large, preconditioning the coefficients and subsequent evaluations is the best way to proceed. By "iterative evaluation", we shall mean that the $(i + 1)$st point is not given until all required evaluations at the ith point are completed. Such a restriction is, of course, quite common. Looking at the proofs of the preconditioning results, it is quite easy to see that for iterative evaluation the following must hold:

1. The evaluation of an nth degree polynomial at k points requires at least $k(\lfloor \frac{1}{2} n \rfloor + 1)$ m/d and kn a/s operations, not accounting operations involving only the coefficients. In particular, if $k = n$, the evaluations require $O(n^2)$ operations.
2. The evaluation of m polynomials of degree n_i $(1 \leqslant i \leqslant m)$ at k points requires at least $k(\lfloor \Sigma_{i=1}^{m} \frac{1}{2} n_i \rfloor + 1)$ m/d and $k\ \Sigma_{i=1}^{m} n_i$ a/s operations, not counting operations involving only the coefficients. In particular, if $n = m = n_i$ for all i, the evaluations require $O(n^3)$ operations.

There are circumstances, however, where all the points of evaluation may be given at once. Problems in approximation theory often satisfy this condition. We should also note that Fourier transforms essentially involve evaluation of nth degree polynomials at $n + 1$ points and subsequent interpolation to recover the coefficients. Of course, in this regard, we are able to choose the points of evaluation.

Let α denote an upper bound on the order of difficulty for matrix multiplication; that is, two $n \times n$ matrices can be multiplied in $O(n^\alpha)$ basic operations. In light of Strassen's [7] fundamental discovery, we know that $2 \leqslant \alpha \leqslant \log_2 7 \approx 2.81$. The following observation is made:

3. n polynomials each of degree n can be evaluated at n points $\{x_1, ..., x_n\}$ in $\leqslant O(n^\alpha)$

$$\begin{pmatrix} a_{0,1} & \cdots & a_{0,1} \\ \vdots & & \\ a_{0,n} & \cdots & a_{0,n} \end{pmatrix} + \begin{pmatrix} a_{1,1} & \cdots & a_{n,1} \\ & a_{i,j} & \\ a_{1,n} & \cdots & a_{n,n} \end{pmatrix} \begin{pmatrix} x_1 & \cdots & x_n \\ & x_j^i & \\ x_1^n & \cdots & x_n^n \end{pmatrix} =$$

$$\begin{pmatrix} \sum_{k=0}^{n} a_{k,1} x_1^k & \cdots & \sum_{k=0}^{n} a_{k,1} x_n^k \\ \vdots & \sum_{k=0}^{n} a_{k,i} x_j^k & \\ \sum_{k=0}^{n} a_{k,n} x_1^k & \cdots & \sum_{k=0}^{n} a_{k,n} x_n^k \end{pmatrix}$$

Since all the powers $\{x_j^i\}$ can be evaluated in $O(n^2)$, and since $\alpha \geqslant 2$, it follows that the required evaluations can be obtained in $O(n^\alpha) \leqslant O(n^{2.81})$ operations.

It is perhaps not quite as obvious how one can use fast matrix multiplication for the evaluation of a single polynomial at several points.

4. An nth degree polynomial can be evaluated at m points $(m \geqslant \sqrt{n})$ in $\leqslant (m/\sqrt{n}) O(n^{\alpha/2}) = m O(n^{(\alpha-1)/2})$. Without loss of generality, we will assume that n is a perfect square. Suppose we want to evaluate $\Sigma_{i=0}^{n} a_i x_j^i$, $1 \leqslant j \leqslant \sqrt{n}$. Let

$$A = \begin{pmatrix} a_1 & a_2 \cdots a_{\sqrt{n}} \\ a_{\sqrt{n}+1} & \cdots \\ \cdots & a_n \end{pmatrix}; \quad X = \begin{pmatrix} x_1 & \cdots x_{\sqrt{n}} \\ x_1^2 & \vdots \\ \vdots & \\ x_1^{\sqrt{n}} & x_{\sqrt{n}}^{\sqrt{n}} \end{pmatrix}$$

Let $Y = AX = (y_{ij})$. Then

$$\sum_{i=0}^{n} a_i x_j^i = a_0 + y_{1j} + \sum_{K=2}^{\sqrt{n}} y_{Kj} x_j^{(K-1)\sqrt{n}}.$$

The evaluation of

$$\{x_j^i \mid 1 \leqslant i, j \leqslant \sqrt{n}\}$$

and

$$\{x_j^{K\sqrt{n}} \mid 2 \leqslant K \leqslant \sqrt{n} - 1\}$$

require only $O(n)$ multiplications. Thus the required evaluations can be accomplished in $\leqslant O((\sqrt{n})^\alpha) + O(\sqrt{n} \cdot \sqrt{n}) + O(n) = O(n^{\alpha/2})$. Therefore, an nth degree polynomial can be evaluated at $\sqrt{n}$ points in $\leqslant O(n^{1.41})$ operations, and at n points in $\leqslant O(n^{1.91})$ operations. In matrix terminology, a row vector $(a_1, ..., a_n)$ can be multiplied by a Vandermonde matrix

$$\begin{pmatrix} x_1 & \cdots & x_n \\ \vdots & & \vdots \\ x_1^n & \cdots & x_n^n \end{pmatrix}$$

in $\leqslant O(n^{1.91})$ operations.

4. Lower bounds and conclusion

It is now appropriate to ask how good are the upper bounds developed in the last section. This question remains open. In the case of n polynomials, each of degree n, evaluated at n points, the best lower bound that we can establish is the lower bound for evaluation at one point; namely n^2 m/d and n^2 a/s operations. In the case of one nth degree polynomial evaluated at m points, we can show the following:

5. It is not difficult to argue that $\Sigma_{i=0}^{n} a_i$, $\Sigma_{i=0}^{n} 2^i a_i$, ..., $\Sigma_{i=0}^{n} m^i a_i$ requires at least $n + m - 1$ a/s and, therefore, this many a/s are required for $\Sigma_{i=0}^{n} a_i x_j^i$, $1 \leq j \leq m$.
6. It is easy to show that there are no non-trivial real vectors $\mathbf{u}$, $\mathbf{v}$ such that

$$\mathbf{u}_{1\times m}\begin{pmatrix} x_1 & \cdots & x_1^n \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ x_m & \cdots & x_m^n \end{pmatrix}\mathbf{v}_{n\times 1} \equiv 0\,.$$

Using a theorem by Fiduccia [2], it follows that $\Sigma_{i=0}^{n} a_i x_j^i$, $1 \leq j \leq m$ requires $m + n - 1$ m/d operations.

Obviously, there is a considerable gap between the provable lower bounds and the achievable upper bounds. We note that even if it should be the case that $\alpha = 2$, our method will only yield an $O(n^{1.5})$ upper bound for $\Sigma_{i=0}^{n} a_i x_j^i$, $1 \leq j \leq n$ in contrast to the best known lower bound $O(n)$. We conclude with another related open problem. Is there an algorithm for exact interpolation which requires less than $O(n^2)$ operations? That is, given $(x_0, y_0), ..., (x_n, y_n)$ can we compute the $\{a_i \mid \Sigma_{i=0}^{n} a_i x_j^i, 0 \leq j \leq n\}$ in less than $O(n^2)$ arithmetics?

References

[1] E.C.Belaga, Some Problems in the Computation of Polynomials, Dokl. Akad. Nauk. SSSR, Vol. 123 (1958).

[2] C.M.Fiduccia, Fast Matrix Multiplication, Conf. Record of Third Annual ACM Symposium on Theory of Computing (1971).

[3] A.M.Garsia, referred to in Knuth, The Art of Computer Programming, Vol. 2, prob 38 in 4.6.4. Addison Wesley, (1969).

[4] J.S.Motzkin, Evaluation of Polynomials and Evaluation of Rational Functions, Bull. Amer. Math. Soc., Vol. 61 (1955).

[5] A.M.Ostrowski, On Two Problems in Abstract Algebra Connected with Horner's Rule, Studies presented to R. Von Mises (Academic Press, New York, 1954).

[6] V.Ya.Pan, Methods of Computing Values of Polynomials, Russian Math. Surveys 21 (1966).

[7] V.Strassen, Gaussian Elimination is Not Optimal, Numerische Mathematik, Vol. 13 (1969).

[8] S.Winograd, On the Number of Multiplications Necessary to Compute Certain Functions, Comm. on Pure and Applied Mathematics, Vol. 23 (1970).